UNIVERSITY OF SOUTHERN QUEENSLAND

SIMULATING THE FLOW OF SOME NON-NEWTONIAN FLUIDS WITH NEURAL-LIKE NETWORKS AND STOCHASTIC PROCESSES

A dissertation submitted by

Trâǹ Cảnh Dũng

B.Eng. (Hon.), Ho Chi Minh City University of Technology, 1982

M.Eng., Haut Alsace University, France, 1995

For the award of

Doctor of Philosophy

December 2003 (Emended July 2004)

To my family

Certification of Dissertation

I certify that the idea, experimental work, results and analyses, software and conclusions reported in this dissertation are entirely my own effort, except where otherwise acknowledged. I also certify that the work is original and has not been previously submitted for any other award.

Dung Tran-Canh, Candidate — Date

ENDORSEMENT

Prof. Thanh Tran-Cong, Principle supervisor — Date

Dr. Ruth Mossad, Associate supervisor — Date

Acknowledgments

I would like to express my deepest appreciation to Professor Thanh Tran-Cong for his effective guidance. Without his continuing support and encouragement this thesis would not have been possible.

In addition, I would like to thank Dr. Ruth Mossad for acting her role as associate supervisor, Ms. Ruth Hilton and Ms. Christine Bartlett (Office of Research and Higher Degrees) for helping to surmount difficult obstacles, Prof. Graham Baker (Dean of the Faculty of Engineering and Surveying, USQ), Dr. David Buttsworth (Postgraduate Studies Coordinator) and Prof. Harry Harris (former Associate Dean, Research and Higher Degrees) for their invaluable support.

I am grateful to Prof. Hans Christian Öttinger (SFIT, Zürich) for his communication and helpful discussions; to the numerous friends and people, whom I have not mentioned by name for their support during the course of my study, both directly and indirectly.

My candidature was at all possible owing to a Postgraduate Scholarship from the University of Southern Queensland and a Scholarship Supplementation from Faculty of Engineering and Surveying. This support is gratefully acknowledged.

Finally, I am indebted to my family, specially my mother, for much support in so many areas over the years and for encouraging me in academic pursuits.

Abstract

The thesis reports a contribution to the development of neural-like network-based element-free methods for the numerical simulation of some non-Newtonian fluid flow problems. The numerical approximation of functions and solution of the governing partial differential equations are mainly based on radial basis function networks. The resultant micro-macroscopic approaches do not require any element-based discretisation and only rely on a set of unstructured collocation points and hence are truly meshless or element-free.

The development of the present methods begins with the use of the multi-layer perceptron networks (MLPNs) and radial basis function networks (RBFNs) to effectively eliminate the volume integrals in the integral formulation of fluid flow problems. An adaptive velocity gradient domain decomposition (AVGDD) scheme is incorporated into the computational algorithm. As a result, an improved feed forward neural network boundary-element-only method (FFNN-BEM) is created and verified. The present FFNN-BEM successfully simulates the flow of several Generalised Newtonian Fluids (GNFs), including the Carreau, Power-law and Cross models. To the best of the author's knowledge, the present FFNN-BEM is the first to achieve convergence for difficult flow situations when the power-law indices are very small (as small as 0.2). Although some elements are still used to discretise the governing equations, but only on the boundary of the analysis domain, the experience gained in the development of element-free approximation in the domain provides valuable skills for the progress towards an element-free approach.

A least squares collocation RBFN-based mesh-free method is then developed for solving the governing PDEs. This method is coupled with the stochastic simulation technique (SST), forming the mesoscopic approach for analyzing viscoelastic fluid flows. The velocity field is computed from the RBFN-based mesh-free method (macroscopic component) and the stress is determined by the SST (microscopic component). Thus the SST removes a limitation in traditional macroscopic approaches since closed form constitutive equations are not necessary in the SST. In this mesh-free method, each of the unknowns in the conservation equations is represented by a linear combination of weighted radial basis functions and hence the unknowns are converted from physical variables (e.g. velocity, stresses, etc) into network weights through the application of the general linear least squares principle and point collocation procedure. Depending on the type of RBFs used, a number of parameters will influence the performance of the method. These parameters include the centres in the case of thin plate spline RBFNs (TPS-RBFNs), and the centres and the widths in the case of multi-quadric RBFNs (MQ-RBFNs).

A further improvement of the approach is achieved when the Eulerian SST is formulated via Brownian configuration fields (BCF) in place of the Lagrangian SST.

The SST is made more efficient with the inclusion of the control variate variance reduction scheme, which allows for a reduction of the number of dumbbells used to model the fluid. A highly parallelised algorithm, at both macro and micro levels, incorporating a domain decomposition technique, is implemented to handle larger problems. The approach is verified and used to simulate the flow of several model dilute polymeric fluids (the Hookean, FENE and FENE-P models) in simple as well as non-trivial geometries, including shear flows (transient Couette, Poiseuille flows), elongational flows (4:1 and 10:1 abrupt contraction flows) and lid-driven cavity flows.

Papers Resulting from the Research

1. Tran-Canh, D., Mai-Duy, N. and Tran-Cong, T. (2000a). Comparison of BEM-FFNN and BEM-RBFN methods for GNF flow analysis. In D.M. Binding, N.E. Hudson, J. Mewis, J-M. Piau, C.J.S. Petrie, P. Townsend, M.H. Wagner and K. Walters (eds). *The XIIIth International Congress on Rheology* (Vol. 2, pp. 140-142). Cambridge: The British Society of Rheology.

2. Tran-Canh, D. and Tran-Cong, T. (2000b). Parallel computation of power-law flows using BEM-NN method and subregion techniques on PVM cluster. The Nation Scientific and Technology center of Hanoi, VietNam. The Conference on high performance scientific computing, 140.

3. Tran-Canh, D. and Tran-Cong, T. (2002a), BEM-NN computation of Generalised Newtonian flows, *J. Engineering Analysis with Boundary Elements*. 26, 15-28.

4. Tran-Canh, D. and Tran-Cong, T. (2002b). Computation of viscoelastic flow using Neural Networks and Stochastic Simulation. *Korea-Australia Rheology Journal*. Vol. 14, No. 2, pp. 161-174

5. Tran-Canh, D. and Tran-Cong, T. (2003a). Meshless simulation of dilute polymeric flows using Brownian Configuration Fields. *Korea-Australia Rheology Journal*, accepted.

6. Tran-Canh, D. and Tran-Cong, T. (2003b). Meshless computation of 2D viscoelastic flows. *Korea-Australia Rheology journal*, submitted.

7. Tran-Canh, D. and Tran-Cong, T. (2003c). Parallel meshfree TPS-RBFN-BCF method for dilute polymer solution. *Korean-Australia Rheology Conference 2003* (pp. 67-70). Gyeongju: The Korean Society of Rheology and The Australian Society of Rheology.

8. Tran-Canh, D., Chu-Quoc, T. and Tran-Cong, T. (2003d). Time-dependent element-free method for dilute polymer solutions, to be submitted.

Contents

List of Frequently Used Acronyms & Abbreviations

AIN	Approximate Identity Networks
AVGDD	Adaptive Velocity Gradient Domain Decomposition
BCF	Brownian Configuration Fields
BDS	Brownian Dynamics Simulation
BEM	Boundary Element Method
BIE	Boundary Integral Equation
CVFNNSS	Computation of Viscoelastic Flows using Neural Networks and Stochastic Simulation
CONNFFESSIT	Calculation of Non-Newtonian Flow: Finite Element and Stochastic Simulation Techniques
CS	Compact support
DD	Domain Decomposition
DE	Differential Equation
FDM	Finite Difference Method
FEM	Finite Element Method
FFNN	Feed Forward Neural Network
FPE	Fokker-Planck Equation
FVM	Finite Volume Method
GNF	Generalized Newtonian Fluids
IE	Integral Equation
MLP	Multilayer Perceptrons
MQ	MultiQuadric
NN	Neural Network
PDE	Partial Differential Equation
RBF	Radial Basis Function
RBFN	Radial Basis Function Network
SDE	Stochastic Differential Equation
SM	Spectral Method
SST	Stochastic Simulation Technique
SVD	Singular Value Decomposition
TPS	Thin Plate Splines

List of Tables

List of Figures

Chapter 1

Introduction

This chapter establishes a motivation for the present research. The chapter provides an overview of the issues and methods associated with the simulation of non-Newtonian flows. Elements of continuum mechanics for macroscopic flow simulations and the kinetic theory in macro-microscopic flow simulations are outlined. A brief review of mesh-free numerical methods, including neural-like networks based numerical methods, is given in contrast with other element-based numerical methods. And finally, the plan of this research is outlined.

1.1 Motivation

In computational rheology, the governing equations can be classified into two different categories. The first group is derived from the fundamental equations which are valid for all materials and describe the universal laws of physics such as the conservation of mass, momentum and energy. The second one focuses on the behavior of materials and is often referred to as either the constitutive equations in the macroscopic approach or microstructure equations using the kinetic theory.

In the conventional macroscopic approach, the requirement of a closed-form constitutive equation is an inherent limitation for complex models that cannot be cast into closed forms in continuum mechanics. The micro-macroscopic approach, which is based on the kinetic theory, has been introduced to bypass this difficulty. In this technique, instead of a closed form constitutive equation, the configuration of a large ensemble of microscopic entities acts as a stress calculator using the Brownian dynamics and kinetic models.

Except for few trivial problems, most engineering problems can only be solved in an approximate manner due to their complexity. Numerical solutions of the governing partial differential equations (PDE) for engineering or scientific problems are based on a wide spectrum of numerical algorithms such as the finite difference method (FDM) (Roache, 1998), the finite element method (FEM) (Reddy and Gartling, 1994; Zienkiewicz and Taylor, 2000a, 2000b), the boundary element method (BEM) (Banerjee and Butterfield, 1981; Bebbria *et al.*, 1984), the finite volume method (FVM) (Patankar, 1980; Versteeg and Malalasekera, 1995), the spectral methods (SM) (Fletcher, 1984). Although the above element-based methods have achieved enormous progress and been efficient numerical methods for various problems in sciences and engineering, they involve extra efforts associated with the discretisation of the analysis domain into a number of finite elements, specially for moving boundary, free surface or three dimensional problems. These disadvantages can be overcome via the mesh-free methods (Atluri and Shen, 2002; Liu, 2003 and Griebel and Schweitzer, 2003) where unknowns are approximated by global functions rather than element-based shape functions.

Recently, neural networks (NNs) have been rapidly applied in many domains of engineering and science (Haykin, 1999) where the feed forward neural networks (FFNNs) consisting of multi-layer perceptron networks (MLPNs) and radial basis function networks (RBFNs) can be considered as accurate approximation schemes (Hornik *et al.*, 1989; Girosi and Poggio, 1990; Park and Sandberg, 1991,1993 and Girosi *et al.*, 1995). These schemes which do not require any element-type discretisation (i.e. fixed connectivity to satisfy a predetermined topology) and rely on a simple set of unstructured discrete collocation nodes in the analysis domain (Kansa, 1990a,b; Zerroukat *et al.*, 1998, 2000; Nguyen-Thien, 1999; He *et al.*, 2000; Tran-Canh *et al.*, 2000a), have proved to be promising.

The motivation for this work is the possibility of an effective and accurate numerical simulation method for non-Newtonian flows that (i) is mesh-free, (ii) does not rely on a closed form constitutive relation, and (iii) is flexible in terms of complex geometry and parallelization.

1.2 Continuum mechanics and macroscopic flow simulations

The macroscopic approach of continuum mechanics makes use of a closed form constitutive equation, connecting the stress and velocity fields, to supplement the mass and momentum conservation laws (Bird *et al.*, 1987a; Barnes *et al.*, 1989; Macosko 1994; Plawsky 2001). This combination yields a set of partial differential (or integro-differential) equations that usually can be solved by means of a suitable numerical method. In this section, the fundamental equations (conservation laws) and constitutive equations are presented.

1.2.1 Fundamental equations

Consider a single-phase continuous medium in a Cartesian coordinate system $x_i, i = 1, 2, 3$. Let u_i be velocity vector; ρ be the density of the material; q_i be the heat flux vector; ϕ be the gravitational potential per unit mass; e be the specific internal energy; σ_{ij} be the Cauchy stress tensor and b_i be the body force per unit mass. The basic physical laws can be expressed as follows

Equation of continuity

$$\frac{D\rho}{Dt} + \rho\frac{\partial u_i}{\partial x_i} = 0, \tag{1.1}$$

where $\frac{D}{Dt}(\centerdot)$ is the material or substantial derivative which gives the time derivative of $(\centerdot)$ associated with a specific fluid element (particle)

$$\frac{D}{Dt}(\centerdot) = \frac{\partial}{\partial t}(\centerdot) + u_i\frac{\partial}{\partial x_i}(\centerdot). \tag{1.2}$$

Equation of motion

The balance of linear momentum yields

$$\rho\frac{Du_i}{Dt} = \rho b_i + \frac{\partial \sigma_{ji}}{\partial x_j}. \tag{1.3}$$

For a steady state and creeping flow, Eq (1.3) reduces to

$$\rho b_i + \frac{\partial \sigma_{ji}}{\partial x_j} = 0. \tag{1.4}$$

The balance of moment of momentum yields

$$\sigma_{ij} = \sigma_{ji}. \tag{1.5}$$

For all materials considered here the stress tensor is assumed symmetric.

Equation of energy

$$\rho\frac{De}{Dt} + \rho\frac{D\phi}{Dt} = -\frac{\partial q_i}{\partial x_i} + \sigma_{ij}D_{ij}, \tag{1.6}$$

where D_{ij} are components of the rate of strain tensor $\mathbf{D}$ and given by

$$D_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right). \tag{1.7}$$

1.2.2 Constitutive equations

The constitutive equation which is derived from experimental observations or theoretical principles (Bird and Wiest, 1995) describes the relation between the stress tensor and the flow kinematics. The constitutive equation can be differential or integral equations. In this work, we are only concerned with the incompressible and isothermal flows. Several well-known constitutive equations for fluids are summarized as follows.

Newtonian fluids

The characteristic of Newtonian fluids is that the stress tensor is linearly proportional to the rate of strain tensor and its constitutive equation is given by

$$\begin{aligned} \boldsymbol{\sigma} &= -p\mathbf{I} + \boldsymbol{\tau}, \\ \boldsymbol{\tau} &= 2\eta_N \mathbf{D}, \end{aligned} \tag{1.8}$$

where p is the hydrostatic pressure that arises due to the incompressibility constraint; $\mathbf{I}$ is the unit tensor; $\boldsymbol{\tau}$ is the extra stress tensor; η_N is the viscosity and $\mathbf{D}$ is the rate of strain tensor.

A fluid is classified as non-Newtonian when its behavior does not obey the above simple linear relation linking the stress tensor and the rate of strain tensor. Following are constitutive equations for some non-Newtonian fluids.

Generalized Newtonian fluids (GNF)

This is the simplest class of non-Newtonian fluids and so called inelastic fluids. The extra stress tensor is given by

$$\boldsymbol{\tau} = 2\eta(\dot{\gamma})\mathbf{D}, \tag{1.9}$$

where $\dot{\gamma} = \sqrt{2\text{tr}(\mathbf{D}^2)}$ is the generalised shear rate with 'tr' is the trace operation. The viscosity η is dependent on the shear-rate. A well-known model for shear-rate-dependent viscosity is the Power-law model whose viscosity is given by

$$\eta = m\dot{\gamma}^{n-1}, \tag{1.10}$$

where n (dimensionless) is the Power-law index and m (with units of $Pa.s$) is a parameter of the model.

Viscoelastic fluids

Among the non-Newtonian fluids, viscoelastic polymeric fluids are the most interesting fluids which have both viscous and elastic characteristics. The behaviour of material, which depends on not only the present stress state but also the deformation history of the material, can be described by the following constitutive equation in general

$$\boldsymbol{\sigma} = -p\mathbf{I} + \boldsymbol{\tau} = -p\mathbf{I} + 2\eta_N\mathbf{D} + \boldsymbol{\tau}^p, \tag{1.11}$$

where the extra stress tensor $\boldsymbol{\tau}$ can be split into the Newtonian-solvent stress $2\eta_N\mathbf{D}$ (which may or may not exist) and the polymer contributed stress $\boldsymbol{\tau}^p$. The extra stress $\boldsymbol{\tau}$ is modelled by several models, from the upper-convected-Maxwell (UCM), Oldroyd-B models to more realistic ones such as the Phan-Thien–Tanner (PTT), Giesekus and Pom-Pom models (Bird *et al.*, 1987a,b; Tanner, 2002). In spite of the difference in complexity as well as in predictive ability, in general, these models can be mathematically represented by an integro-differential equation of the following form

$$\boldsymbol{\tau} = \int_{-\infty}^{t} m(t-s)\mathbf{S}_t(s)ds, \tag{1.12}$$

where $\mathbf{S}_t(s)$ is the deformation-dependant tensor; m is the memory function, or by a differential equation of the form

$$\lambda_H \frac{\Delta\boldsymbol{\tau}}{\Delta t} + f(\boldsymbol{\tau}, \mathbf{D}) = 2\eta\mathbf{D}, \tag{1.13}$$

where $\frac{\Delta}{\Delta t}(\centerdot) = \frac{\partial}{\partial t}(\centerdot) + \mathbf{u}\cdot\nabla(\centerdot) - \boldsymbol{\kappa}(\centerdot) - (\centerdot)\boldsymbol{\kappa}^T$ is the upper-convected time derivative; $\boldsymbol{\kappa}$ is the velocity gradient; η is a viscosity of the fluid; f is the model-dependent function; λ_H is a relaxation time. The form of f and possibly the variation of λ_H and η leads to a variety of models as follows.

• If $f = \boldsymbol{\tau}$, and λ_H and η are constants, we have the case of the upper-convected Maxwell model. The additional contribution of a Newtonian solvent $(2\eta_N\mathbf{D})$ to the stress leads to the Oldroyd-B model.

• If f is a function of $\mathrm{tr}(\boldsymbol{\tau})$ and λ_H and η are constants, we have various special PTT models (Phan-Thien and Tanner 1977; Phan-Thien 1978, 1984; Tanner 2002). As a further modification, a slip coefficient ξ may be used in PTT models in which a term $\xi(\mathbf{D}\boldsymbol{\tau} + \boldsymbol{\tau}\mathbf{D})$ is added to the left hand side of (1.13), we have

1. If $f = 1 + \frac{\lambda\epsilon}{\eta}\mathrm{tr}(\boldsymbol{\tau})$, we have the linear PTT model.
2. If $f = \exp\frac{\lambda\epsilon}{\eta}\mathrm{tr}(\boldsymbol{\tau})$, we have the exponential PTT model.

1.3 Kinetic theory and micro-macroscopic flow simulations

Although a very large percentage of the research in computational flow simulation has been based on the macroscopic approach, there appears to be an alternative. In fact, the main concern in the macroscopic approach is to find appropriate constitutive equations that can be applied to the increasing numbers of complex fluids and to estimate the impact of a closure hypothesis made to build suitable constitutive equations on the quality of the final result. The difficulty of making a closure hypothesis can be overcome by the micro-macroscopic (or mesoscopic) approach. This approach is developed on the basis of kinetic theory and consists of finding a formula of the macroscopic stress tensor in term of the microscopic dynamics of the polymer chains and treating explicitly both scales in the simulation.

Hence, when using the macroscopic approach, the strain-history-dependent stresses are traditionally obtained from constitutive equations (1.13) or (1.12), whereas in the micro-macroscopic approach, the equation system (1.1) and (1.3) has to be treated together with

- The Fokker-Planck equation (FPE), which is given by (in the absence of external forces) (Öttinger, 1996)

  $$\frac{\partial}{\partial t}\psi(\mathbf{Q},t) = -\frac{\partial}{\partial \mathbf{Q}} \cdot [\mathbf{A}(\mathbf{Q},t)\psi(\mathbf{Q},t)] + \frac{1}{2}\frac{\partial}{\partial \mathbf{Q}} \cdot [\frac{\partial}{\partial \mathbf{Q}} \cdot \{\mathbf{L}(\mathbf{Q},t)\psi(\mathbf{Q},t)\}], \tag{1.14}$$

  where $\psi(\mathbf{Q},t)$ denotes the probability density function of the configuration vector $\mathbf{Q}$ (polymer configuration) at the time t; $\mathbf{A}(\mathbf{Q},t)$ is a d-component column vector; $\mathbf{L}(\mathbf{Q},t)$ is a positive semi-definite $d \times d$ matrix; $\mathbf{A}$ and $\mathbf{L}$ define the deterministic and statistical components of the model, respectively; and

- An expression for the stress tensor involving the polymer configuration $\mathbf{Q}$ whose distribution is given by ψ.

1.3.1 Numerical analysis of micro-macroscopic flow simulations

Although the micro-macroscopic approach uses the full information of the stresses which is contained in the configuration of the molecules, it is not used to simulate the largest system possible but only to combine a macroscopic discretization with the microscopic simulation of as few polymer molecules as necessary to achieve a given statistical accuracy (Van de Brule and Hoogerbrugge, 1995;

Öttinger, 1996; Lielens, *et al.*, 1999; Binous and Phillips, 1999). The simulations based on the coupling of the macroscopic equations and the microstructural equations are considered as hybrid simulations. Depending on the kind of the microstructural equation used, there are various numerical schemes in the mesoscopic approach. In this section, the hybrid simulation using direct solution of the FPE and the hybrid simulation using stochastic simulation technique (SST) are presented.

Mesoscopic simulation using direct solution of the FPE

As the name suggests, this scheme is based on a direct solution of the FPE and the FEM for computing complex flows that can be modelled with a small number of degrees of freedom. The direct simulations using FPE are usually based on weighted residual methods in which a set of basis functions is employed. The basis functions can be chosen from the eigenfunctions of the FPE. The simulation based on this approach has been applied to the dilute, multi-bead-rod model (Fan, 1989) and the liquid crystalline polymers using the Doi model (Nayak, 1998). Fan (1989) employed the Galerkin method with spherical harmonic basis functions to discretise the configuration space and the BEM for the physical space. Nayak (1998) used the discontinuous Galerkin method in conjunction with wavelet analysis to simulate the dynamics of the rigid dumbbell and Doi models. A review on this area can be found in Suen *et al.* (2002).

Mesoscopic simulation using SST

The first mesoscopic simulation using SST, namely the CONNFFESSIT (Calculation of Non-Newtonian Flow: Finite Element and Stochastic Simulation Techniques), was proposed by Laso and Öttinger (1993) and developed for the start-up problem of simple shear flows. This scheme has dominated micro-macroscopic simulations.

Instead of solving the FPE directly to get the stress tensor, the authors proposed a simulation algorithm that couples stochastic simulation technique of a kinetic model and the numerical solution of the fundamental equations of the conservation laws. In the last ten years, many works concerning this approach have appeared (Feigl *et al.*, 1995; Hua and Schieber, 1998; Halin *et al.*, 1998; Bell *et al.*, 1997; Hulsen *et al.*, 1997; van den Brule *et al.*, 1997; Öttinger *et al.*, 1997; Bonvin *et al.*, 1999; Fan *et al.*, 1999; Wapperom *et al.*, 2000, Somasi et Khomami, 2000; Jourdain *et al.*, 2002; Tran-Canh and Tran-Cong, 2002b).

The works based on CONNFFESSIT have focused on either applying the CONNFFESSIT to solve typical viscoelastic problems (Laso *et al.*, 1997; Cormenzana *et al.*, 2001) or modifying the algorithm to improve the accuracy and efficiency of the method (Öttinger *et al.*, 1997; Hulsen *et al.*, 1997; Halin *et al.*, 1998; Wapperom *et al.*, 2000). While the majority of these works are based on the coupling of FEM and SST, Bell *et al.* (1997) employed a spectral method coupled with Brownian dynamics simulation, which provides the advantage of a high convergence rate. Recently, Tran-Canh and Tran-Cong (2002b, 2003a) coupled the RBFN-based meshless method with SST to solve several viscoelastic flow problems using single dumbbell models. Since the introduction of CONNFFESSIT, this approach, and its variants such as the Brownian configuration fields (BCF) method (Eulerian CONNFFESSIT) (Hulsen *et al.*, 1997), prove to be a powerful and efficient numerical method for simulating the flow of complex fluids.

1.4 A brief review of the conventional numerical methods

In order to obtain the solution of a problem, a set of governing equations, usually PDEs, must be solved given certain boundary conditions. Exact solutions are often possible only for simple problems. However, most engineering problems involve complex geometry and complex material and/or geometrical nonlinearities. Such practical problems can only be solved by numerical methods. A numerical solution requires the discretisation of the variable fields. The object of the discretisation is to reduce the governing PDEs of the problem to a set of discrete simultaneous equations in terms of a finite number of nodal variables, most of which are unknowns to be found. This set of equations can be then solved advantageously on a computer. Some principal numerical methods such as FEM, BEM, FVM, FDM and SMs are presented briefly in this section. With scarcely an exception, these techniques can be considered as variants of the weighted residual methods (Brebbia *et al.*, 1984).

Consider the following equations

$$\mathcal{L}(u_o) = y \qquad \text{in} \quad \Omega, \tag{1.15}$$

with boundary conditions

$$\begin{aligned} \mathcal{S}_1(u_o) &= b_1 \qquad \text{in} \quad \partial\Omega_1, \\ \mathcal{S}_2(u_o) &= b_2 \qquad \text{in} \quad \partial\Omega_2, \end{aligned} \tag{1.16}$$

where Ω is the domain under consideration; $\partial\Omega = \partial\Omega_1 \cup \partial\Omega_2$ is the boundary of Ω; $\mathcal{L}$, $\mathcal{S}_1$, $\mathcal{S}_2$ are some differential operators; y is a given function; b_1 and b_2 are the Dirichlet and Newmann conditions on $\partial\Omega_1$ and $\partial\Omega_2$, respectively and u_o is the exact solution of the problem. u_o can be approximated by u as follows

$$u_o \approx u = \sum_{i=1}^{m} w^i h^i + w^o, \tag{1.17}$$

where $\{h^i(\mathbf{x})\}_{i=1}^{m}$ is the set of linearly independent functions, so called nodal functions, and chosen to satisfy certain given conditions; $\mathbf{x}$ is the spatial coordinates of points in the domain Ω; $\{w^i\}_{i=1}^{m}$ are unknowns coefficients and w^o is included to satisfy the nonhomogeneous part of the boundary conditions. In some cases, w^i can be associated with nodal values of the considered variables. Although the solution u is considered as a scalar, it could be a vector $u_i, (i = 1, 2, 3)$.

If the approximant u satisfies all the boundary conditions of the problem, the substitution of the approximation for u_o into Eq (1.15) produces a residual or error function R as follows

$$R = \mathcal{L}(u) - y \neq 0 \qquad \text{in} \quad \Omega. \tag{1.18}$$

If u does not satisfy all boundary conditions, there are other types of residual functions as follows

$$\begin{aligned} R_1 &= \mathcal{S}_1(u) - b_1 \neq 0 \qquad \text{in} \quad \partial\Omega_1, \\ R_2 &= \mathcal{S}_2(u) - b_2 \neq 0 \qquad \text{in} \quad \partial\Omega_2. \end{aligned} \tag{1.19}$$

The aim of the numerical methods is to make these errors as small as possible over the domain and on the boundary. The way in which the errors R, R_1 and R_2 are distributed over the domain and on the boundary produces different types of approximate methods.

1.4.1 Terminology

In the following discussion, by 'element' we mean a predefined topology or connectivity involving a number of nodes. For example, a linear triangular element will involve three non-collinear nodes connected either in a clockwise or anti-clockwise manner. An analysis domain can then be discretised by an appropriate number of these elements. The collection of these elements, with proper relation between them, is called a mesh. Meshing (and re-meshing) is used to denote element-based discretisation. In contrast, element-free or mesh-free or mesh-less dicretisation does not involve elements as defined above. Instead, element-free discretisations normally rely on a set of unstructured discrete points.

1.4.2 Finite difference method (FDM)

The method describes the unknowns of the problem by sampling appropriate variables at the nodal points generated by a grid of co-ordinate lines. The differential governing equations are discretized into a system of algebraic equations by means of local expansions of unknowns at grid points based on truncated Taylor series expansion. The set of algebraic equations which denotes the relation between the nodal values and the neighboring nodal values can be solved by an iterative method such as the Newton-Raphson scheme. FDM has been widely employed in a large range of problems in continuum mechanics (Gatski and Lumley, 1978; Adachi *et al.*, 1978; Peyret and Taylor, 1883 and Crochet *et al.*, 1984). However, the large truncation error for non-linear problems and the difficulties of handling arbitrary geometry limit the range of their application. More details about the method can be found in Crochet *et al.* (1984) and Smith (1978).

1.4.3 Finite element method (FEM)

In this method, the domain is subdivided into a number of non-overlapping finite elements of predefined topology. A set of piecewise functions, usually polynomials, is employed to describe the local variations of the unknowns. The original differential equations are transformed into integral equations by applying an appropriate method such as the weighted residual technique or variational principle. As a result we obtain a set of algebraic equations for the unknown coefficients of the approximating functions. The theory of FEM has been developed initially for structural stress analysis. It has now become a popular method in computational rheology (Thomaset, 1981, Crochet *et al.*, 1984). Extensive works have applied to the area of viscoelastic flows (Kawahara and Takeuchi, 1977; Crochet and Keunings, 1980; Beris *et al.*, 1986). For strong convection problems, a number of upwinding schemes have been developed such as the streamline upwinding (Marchal and Crochet, 1987) and the streamline upwinding Petrov-Galerkin methods (Brooks and Hughes, 1982). Many different advanced techniques have been proposed for an efficient solution of crucial problems in viscoelastic fluid flows such as the problem of high Weissenberg number: the elastic viscous split stress (Perera and Walters, 1977), and the adaptive viscoelastic stress splitting scheme (Sun *et al.*, 1996). Besides successes, FEM still has some drawbacks. Among these is the problem of discretisation which is required on the whole domain. This can be difficult and may also be time consuming especially for moving boundary or three dimensional problems.

1.4.4 Boundary element method (BEM)

The method is based on solving a set of integral equations on the boundary of the domain in which the governing PDEs are converted into equivalent integral equations via several mechanisms such as the reciprocal theorems combined with the relevant Green function, or the method of weighted residual. The dimensionality of the problem in BEM is reduced by one, which is considered as an advantage of the BEM in comparison with other numerical methods for certain large scale and three dimensional problems, where the demand in computing resource is very high. The BEM has been developed and applied rapidly in a wide range of complex engineering problems (Banerjee and Butterfield, 1979; Banerjee and Watson, 1986; Brebbia *et al.*, 1984). In computational rheology, many complex fluid flow problems have been successfully solved by BEM such as the squeeze-film flows (Phan-Thien *et al.*, 1987), the extrusion problems (Tran-Cong and Phan-Thien, 1988). However while BEM works very well for the linear and mildly nonlinear problems, it is less successful with highly nonlinear viscoelastic problems (Tanner and Xue, 2002).

1.4.5 Finite volume method (FVM)

The FVM method was originally developed as a special finite difference formulation in which the analysis domain is partitioned into finite non-overlapping control volumes. The weighted functions h^i, $(i = 1, 2, ..., n)$ are chosen so that they become unity for certain control volume $\Delta\Omega$ and zero for the rest of the control volumes. The fundamental statement of FVM over a control volume $\Delta\Omega$ is given by

$$\int_{\Delta\Omega} (\mathcal{L}u - y) d\Omega = 0, \tag{1.20}$$

where $\mathcal{L}$ is a differential operator and y is a given function. The conservation laws are applied over a control volume by implementing integration instead of using a truncated Taylor series expansions to the differential governing equations as in FDM. Details of the method and its applications can be found in Patankar (1980); Versteeg and Malalasekera (1995); and Tanner (2002).

1.4.6 Spectral methods (SM)

Spectral methods approximate the unknowns by means of truncated Fourier series or series of Chebyshev polynomials. Unlike the finite difference or finite element approaches, the approximation of spectral methods is valid on the whole computation domain. Consider the heat diffusion problem which is governed by the following equation

$$\frac{\partial u}{\partial t} = \mathcal{L}u + g(x, t), \tag{1.21}$$

where $\mathcal{L}$ is a linear spatial differential operator and g is a given function. The solution u can be approximated by u_m in the truncated series form as follows

$$u_m(x, t) = \sum_{i=1}^{m} w^i(t) h^i(x), \tag{1.22}$$

where h^i's are linearly independent functions which can be Fourier or Chebyshev polynomial series. With appropriate boundary and initial conditions, the application of the weighted residual method to Eq (1.21) leads to a system of algebraic equations for the unknown coefficients w^i of Fourier or Chebyshev series. More detail on the method can be found in Gottlieb and Orszag (1977). The method has been employed to solve successfully various problems in computational fluid mechanics (Beris *et al.*, 1992; Talwar *et al.*, 1994; Van Kemenade and Deville, 1994). The advantages of spectral methods lie in their high degree of convergence (Beris *et al.*, 1987) and accuracy of solutions without using any upwind scheme to ensure the stability (Talwar *et al.*, 1994). However, the method is sensitive to the geometry of problems.

1.5 Element-free numerical methods

Although the element-based numerical methods (FEM, BEM, FVM) are efficient and well-established, they suffer from drawbacks such as tedious meshing, re-meshing in problems involving complexity such as large deformation, crack propagation, moving boundary, especially in three dimensions. Spectral methods are more accurate, but have severe restrictions on the geometry where the domain must be regular to obtain the tensor product mesh (Kansa and Hon, 2000). Due to these problems, element-free methods for boundary value problems have recently attracted much attention (Atluri and Shen, 2002; Liu, 2003) from researchers. For the last several decades, many different element-free methods have been proposed and developed: the smooth particle hydrodynamics method (Gingold and Monaghan, 1977); the meshless collocation method using multi-quadric radial basis function (Kansa, 1990a,b); the diffuse element method (Nayroles *et al.*, 1992); the element-free Galerkin method (Belytschko *et al.*, 1994); the reproducing kernel particle method (Liu *et al.*, 1996); hp-cloud method (Duarte and Oden, 1996); the partition of unity finite element method (Babuska and Melenk, 1997); the meshless local Petrov-Galerkin method (Atluri *et al.*, 1999). Among these, the methods which use shadow elements for the integration of the weak-form are not truly meshless. The differences in these

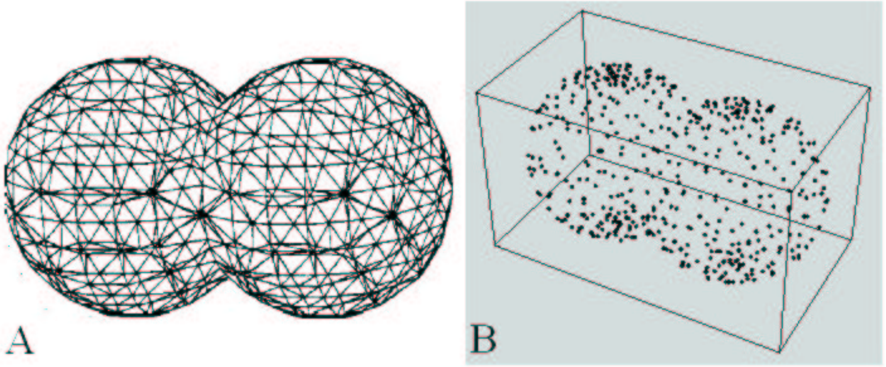


Figure 1.1: An example of domain discretisations using the element-based and element-free methods: A) Element-based method; B) Element-free method. The picture is taken from Chen *et al.* (2003).

schemes stem from the techniques used for interpolating the trial function. The main advantage of the meshless methods is to alleviate or eliminate the extra effort of meshing and re-meshing in the entire domain under consideration (Fig. 1.1), and other drawbacks associated with the element-based methods, such as element distortion, locking, etc. A very good survey of element-free methods can be found in Belytschko *et al.* (1996); Atluri and Shen (2002) and Liu (2003). In the present work, the truly meshless method based on neural-like networks, for example RBFNs, is employed for function approximation and for the numerical solution of PDEs. The method based on Kansa's algorithm has attracted much attention in both theory and application (Franke, 1982; Frank and Schaback, 1998; Zeroukat *et al.*, 1998, 2000). The neural-like network-based element-free methods employed in this study need more specific discussion and are deferred to an appropriate location in subsequent chapters.

1.6 Outline of the present research

Neural-networks, especially RBFNs, for approximating functions and solving PDEs will be presented. The present work aims to couple the MLPNs and

RBFNs with other numerical methods for solving non-Newtonian fluid flow problems: (i) use FFNNs in conjunction with the BEM to solve the generalized Newtonian fluid (GNF) flow problems and (ii) use RBFN-based mesh-free methods in combination with the power and generality of Brownian dynamic techniques from the Lagrangian and Eulerian point of view to solve the viscoelastic fluid flow problems, particularly, the flow of dilute polymer solutions.

The present thesis is organized as follows:

• Chapter 2 consists of two parts concerning the basic tools used in the present work. The first one is to describe the deterministic and stochastic numerical methods such as integral equation (IE), feed-forward neural-like networks and stochastic differential equations (SDEs), which are employed as the basis of the proposed procedures. The second is to review the Brownian dynamics technique and the modelling of dilute polymer solution using single dumbbell models.

• Chapter 3 reports on the method of coupling FFNNs (which are considered as an efficient approximation of functions and their derivatives) and the boundary element method. This chapter will not detail the approximation of functions and their derivatives but uses FFNNs in conjunction with the boundary element method for the numerical solution of GNF flows (Tran-Canh and Tran-Cong, 2002a). Some comparisons between MLPNs and RBFNs are also reviewed briefly (Tran-Canh *et al.*, 2000a).

• Chapter 4 presents a new technique (the Computation of Viscoelastic Flow using Neural Network and Stochastic Simulation (CVFNNSS)) coupling the powerful stochastic simulation technique with a RBFN-based element-free method. The method is built up from the Lagrangian point of view and the implementation is carried out for the transient shear viscoelastic flow problems (Tran-Canh and Tran-Cong, 2002b).

• Chapter 5 is to concentrate on developing CVFNNSS for the case of two dimensional problems. Owing to the mesh-free feature, it is possible to advance the concept of flexible effective stress averaging volume where overlapping volumes can be used to ensure a statistically sufficient number of dumbbells around a particular point to calculate the stress at that point (Tran-Canh and Tran-Cong, 2003b). Some test and benchmark flow problems of dilute polymer solutions (the steady planar Poisseuille and steady 4:1 axisymmetric contraction flows) are simulated to verify the method.

• Chapter 6 is to develop further the present method in which the RBFN-based mesh-free method is combined with the hybrid simulation approach from the Eulerian point of view (also called the Brownian configuration fields) instead of the Lagrangian point of view. The application of the Brownian configuration fields in conjunction with the variance reduction technique allows for a remarkable reduction of the noise associated with the stochastic process, statistical errors and the computation time. The concept of Brownian configuration fields also eliminates the need and effort of the particle tracking. Some test and benchmark problems, namely the Couette, Poiseuille and the lid driven cavity flows, are carried out to demonstrate the working of the method (Tran-Canh and Tran-Cong, 2003a).

• Chapter 7 reports on domain decomposition and parallelization techniques in conjunction with the present Eulerian CVFNNSS. The iterative non-overlapping domain decomposition technique is employed in both macro and microscopic components of the method. The 10:1 planar contraction and lid square driven cavity flows are used to estimate the efficiency of the domain decomposition and parallization techniques (Tran-Canh and Tran-Cong, 2003c).

• Chapter 8 gives some concluding remarks emanating from this research and some recommendations for further works.

Chapter 2

Fundamental background

In this chapter, several basic tools, which are the background of the computation and the polymer dynamics used in the present work, are described. These tools are the boundary integral equations (BIEs), feed-forward neural-networks (FFNNs), stochastic differential equations (SDEs) and Brownian dynamic simulations. Other tools and techniques will be introduced later in the relevant chapters. In this dissertation, the FFNNs, especially radial basis function networks (RBFNs) which are developed as an element-free numerical method for function approximation and numerical solution of partial differential equations (PDEs), will be coupled with other methods in both macroscopic and mesoscopic approaches. The coupling with the boundary element method is for the numerical solution of the boundary integral equation in which FFNNs play the role of an approximator to facilitate the finding of a particular solution of the problem. In the hybrid simulation using stochastic simulation technique (SST), RBFNs are used as a truly mesh-free method for the approximation of a given function and the solution of PDEs.

2.1 Boundary Integral Equations

The formulation of BIEs which is employed in this work is derived directly from the governing PDE (1.4) while the indirect formulation which is obtained from the potential theory, can be found, for example, in Banerjee and Butterfield (1981) and Brebbia *et al.* (1984). Although the current formulation is valid for different classes of material, it is here described for incompressible fluids only. In general, the stress tensor of a body in an equilibrium configuration can be written as

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}^l + \boldsymbol{\tau}^p, \tag{2.1}$$

where $\boldsymbol{\sigma}^l$ and $\boldsymbol{\tau}^p$ are a fictitious linear part and the remaining non-linear part, respectively. The total traction corresponding to the stress tensor is

$$\begin{aligned} \mathbf{t} &= \mathbf{n} \cdot (\boldsymbol{\sigma}^l + \boldsymbol{\tau}^p) \quad \text{or} \\ t_j &= \sigma_{jk} n_k = \sigma^l_{jk} n_k + \tau^p_{jk} n_k, \end{aligned} \tag{2.2}$$

where $\mathbf{n}$ is the outward unit normal to the boundary. For the Newtonian fluids, the linear stress $\boldsymbol{\sigma}^l$ (also $\boldsymbol{\sigma}^N$ elsewhere in this work) is given in Eq (1.8) as follows

$$\boldsymbol{\sigma}^l = -p\mathbf{I} + 2\eta_N \mathbf{D}. \tag{2.3}$$

For viscoelastic fluids, comparing Eq (1.11) with Eq (2.1), we have

$$\boldsymbol{\tau} = 2\eta_N \mathbf{D} + \boldsymbol{\tau}^p, \tag{2.4}$$

where $2\eta_N \mathbf{D}$ and $\boldsymbol{\tau}^p$ are the contribution of the linear and non-linear components, respectively, to the extra stress. From Eq (2.1), the equation of motion for steady state (1.4) is given by

$$\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}^l + (\boldsymbol{\nabla} \cdot \boldsymbol{\tau}^p + \mathbf{b}) = 0, \tag{2.5}$$

where $\boldsymbol{\sigma}^l$ is determined from Eq (2.3). Assuming that $\boldsymbol{\nabla} \cdot \boldsymbol{\tau}^p$ is available, $(\boldsymbol{\nabla} \cdot \boldsymbol{\tau}^p + \mathbf{b})$ can be considered as a pseudo-body force. Applying the method of weighted residuals, Eq (2.5) can be written as follows (Brebbia *et al.*, 1984)

$$\int_\Omega \left(\frac{\partial \sigma^l_{jk}}{\partial x_k} + \left(b_j + \frac{\partial \boldsymbol{\tau}^p_{jk}}{\partial x_k} \right) \right) u^*_j d\Omega = 0, \tag{2.6}$$

where $\mathbf{u}^*$ is an arbitrary velocity weighting field and Ω is the domain under consideration with boundary Γ. Let $(\mathbf{u}, \boldsymbol{\sigma}^l)$ and $(\mathbf{u}^*, \boldsymbol{\sigma}^{l*})$ be two equilibrium states of a material characterized by Eq (2.3), the following identity is given by the reciprocity principle

$$\int_\Omega \sigma^l_{jk} D^*_{jk} d\Omega = \int_\Omega \sigma^{l*}_{jk} D_{jk} d\Omega, \tag{2.7}$$

where D_{jk} and D^*_{jk} are determined by Eq (1.7). Assuming that the desired boundary conditions are satisfied exactly, and using the identity (2.7), Eq (2.6) can be recast as follows (more detail can be found in Tran-Cong, 1989 and Becker, 1992).

$$\begin{aligned} C_{ij}(\mathbf{x})u_j(\mathbf{x}) = \int_\Gamma u^*_{ij}(\mathbf{x},\mathbf{y})t_j(\mathbf{y})d\Gamma(\mathbf{y}) - \int_\Gamma t^*_{ij}(\mathbf{x},\mathbf{y})u_j(\mathbf{y})d\Gamma(\mathbf{y}) \\ + \int_\Omega \left(u^*_{ij}(\mathbf{x},\mathbf{y})b_j - \tau^p_{jk}(\mathbf{y}) \frac{\partial u^*_{ij}(\mathbf{x}),\mathbf{y}}{\partial x_k} \right) d\Omega(\mathbf{y}), \end{aligned} \tag{2.8}$$

where $\mathbf{x}, \mathbf{y} \in \Omega$; $u_j(\mathbf{y})$ is the j-component of the velocity at a point $\mathbf{y}$; $t_j(\mathbf{y})$ is the j-component of the total boundary traction at $\mathbf{y}$; $\tau^p_{jk}(\mathbf{y})$ is the jk-component of $\boldsymbol{\tau}^p$ at $\mathbf{y}$; $u^*_{ij}(\mathbf{x},\mathbf{y})$ is the i-component of velocity field at $\mathbf{x}$ due to a point force in j-direction at $\mathbf{y}$ (Kelvin fundamental solution, see Appendix A.1) and $t^*_{ij}(\mathbf{x},\mathbf{y})$ is its associated traction. $C_{ij}(\mathbf{x})$ depends on the local geometry, for example $C_{ij}(\mathbf{x}) = \delta_{ij}$ (δ_{ij} is the Kronecker delta) if $\mathbf{x}$ lies in Ω and $C_{ij}(\mathbf{x}) = \frac{1}{2}\delta_{ij}$ if $\mathbf{x}$ is located on the boundary Γ and the surface at $\mathbf{x}$ is Liapunov smooth. More detail can be found in Tran-Cong (1989) and Phan-Thien and Kim (1994).

The volume integral on the RHS of Eq (2.8) can be determined by several techniques, each of which have advantages as well as drawbacks. Some of these techniques are briefly described below.

1. The use of Fourier expansion (Tang, 1988). In this scheme, the body force is expressed by a set of functions using Fourier expansions. The set of volume integrals is transformed into boundary integrals by using Green's second identity.

2. The multiple reciprocity method (Novak and Brebbia, 1989). The basic idea

of this method is to employ a set of higher order fundamental solutions which permit the application of the reciprocity theorem. The main drawback of this scheme is that it cannot be easily applied to general non-linear problems.

3. The Dual reciprocity method (Partridge *et al.*, 1992) or the particular solution method (Zheng *et al.*, 1991; Coleman *et al.*, 1991 and Ingber and Phan-Thien, 1992). In this technique, an approximate particular solution, which is obtained by expressing the integrand in terms of a linear combination of RBFs, is employed to transform the volume integrals into the boundary.

4. Cell integration scheme. In this approach, the volume under consideration is divided into a series of internal cells. On each of them, a numerical scheme such as Thomson or Gauss quadrature can be applied. The method requires the discretisation of domain and the introduction of further approximations which diminishes the attractiveness of the BEM.

5. The Monte Carlo method. This scheme requires a system of random integration points rather than the regular integration grid used in the Cell integration approach.

2.2 Feed-forward neural networks

Recently, many NN-models have been proposed to solve various problems in different disciplines such as physics, engineering and computer science. Among the NN-models, the MLPNs and RBFNs have emerged as the leading classes. As pointed out by Poggio and Girosi (1990), the ability of learning in a FFNN is closely related to its approximation capabilities. The capability of FFNNs in approximating arbitrary input-output mappings has been demonstrated earlier in the works of Hornik *et al.* (1989) and Cybenko (1989) for MLPNs and in the work of Park and Sandberg (1991) for RBFNs. While both classes have been considered as ‘universal approximators’ and met with remarkable success in applications, there are some fundamental differences between the theory and practice of applying and designing these networks (Van-Yee, 1998; Haykin, 1999). In the present works, both MLPNs and RBFNs will be described as basic tools for function approximation, but only the RBFN is employed as a meshless method to solve PDEs.

In general, a FFNN $f(\mathbf{x}) : \mathbf{R}^d \rightarrow \mathbf{R}$ can be represented by the following summation

$$f(\mathbf{x}) = \sum_{j=1}^{m} w^j h^j(\mathbf{x}, \mathbf{c}^j, b^j), \tag{2.9}$$

where $\mathbf{x} = (x_1, x_2, ..., x_d)^T \in \mathbf{R}^d$; $\{w^j\}_{j=1}^m$ is a set of weights which are adjusted in the training phase; $\mathbf{c}^j \in R^d$ and $b^j \in R$ are the parameters of neural networks. The form of functions h^j's depends on the specific class of considered networks.

2.2.1 Brief review of MLPNs

MLPNs have been applied successfully to solve some difficult and diverse problems by training them in a supervised manner with a popular algorithm known as the back-propagation algorithm which is based on the error-correction learning rule (Rojas, 1996 and Haykin, 1999). Some significant distinctive features of MLPNs are (i) the non-linear activation function at each neuron is smooth (i.e. differentiable everywhere). The presence of non-linearities has an important meaning because otherwise the input-output mapping of the NN can be reduced to that of single-layer perceptron; (ii) the hidden layers enable MLPNs to learn complex tasks by extracting progressively more meaningful features from the input training patterns; (iii) the high degrees of connectivity are determined by the population of synaptic connections or their weights.

For MLPNs, the function h^j of Eq (2.9) is given by

$$h^j(\mathbf{x}, \mathbf{c}^j, b^j) = \phi((\mathbf{c}^j)^T\mathbf{x} + b^j), \tag{2.10}$$

where ϕ is a mapping $R^d \rightarrow R$; $\mathbf{c}^j \in R^d$ is a 'direction vector' and b^j is a 'threshold'. The continuously differentiable nonlinear activation functions commonly used in MLPNs are the logistic and hyperbolic tangent function given by respectively

$$\begin{aligned} h^j(\mathbf{x}, \mathbf{c}^j, b^j) &= \frac{1}{1+\exp(-((\mathbf{c}^j)^T \cdot \mathbf{x} + b^j))}, && (2.11) \\ h^j(\mathbf{x}, \mathbf{c}^j, b^j) &= b^j \tanh((\mathbf{c}^j)^T \cdot \mathbf{x}). && (2.12) \end{aligned}$$

The logistic sigmoid function is considered as the best for MLPNs trained by the back-propagation algorithm (Rojas, 1996; Haykin, 1999). After the training process is successfully completed the set of weights corresponding to the chosen architecture is obtained.

2.2.2 Radial basis function networks

Network description

The RBFN is the principal NN-model employed in the present work. An important property of the RBFN is that it is considered as a linear combination of the radial basis functions in a single hidden layer architecture where the function h^j of Eq (2.9) is a radial basis function and can be written as follows

$$h^j(\mathbf{x}, \mathbf{c}^j, b^j) = h^j(r, b^j) = \phi\left(\frac{r}{b^j}\right), \qquad (2.13)$$

where ϕ is a mapping $R^d \rightarrow R$; $\mathbf{c}^j \in R^d$ and b^j are the centre vector and spread parameter (also namely the width) of the jth node, respectively and $r = \parallel \mathbf{x} - \mathbf{c}^j \parallel$ while $\parallel \centerdot \parallel$ is the Euclidean norm. Figure (2.1) shows a fully connected RBFN with a single hidden layer. The linear combination of nonlinear radial basis functions is the key to the RBFN's representational ability while maintaining the computational and analytical tractability. The final (output) layer, fully connected with the hidden layer, is composed of a linear unit whose weights are considered as unknowns of the neural network.

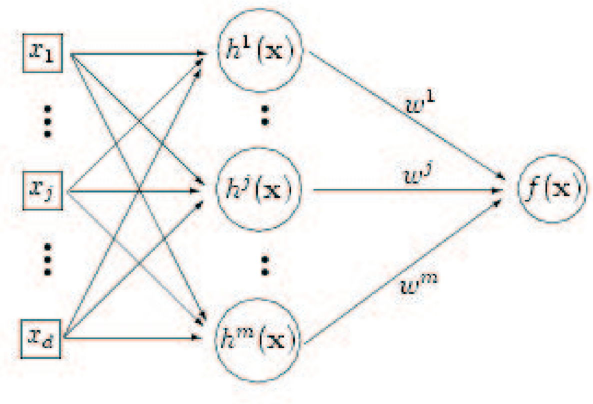


Figure 2.1: Architecture of a fully connected radial basis function network. $\mathbf{x} \in R^d$; $\{w^j\}_{i=1}^m$ is the set of weights of the NN. Each of d components of $\mathbf{x}$ feeds forward to m basis function neurons whose outputs are linearly combined with weights into the network output $f(\mathbf{x})$.

Some well known radial basis functions

Among radial basis functions, the following are of particular interest in applying RBFNs

- Gaussian-RBF

$$h^j(r, b^j) = h^j(\|\mathbf{x} - \mathbf{c}^j\|, b^j) = \exp\left(-\frac{r^2}{(b^j)^2}\right), \tag{2.14}$$

- Multiquadrics RBF (MQ-RBF)

$$h^j(r, b^j) = h^j(\|\mathbf{x} - \mathbf{c}^j\|, b^j) = \sqrt{r^2 + (b^j)^2}, \tag{2.15}$$

- Inverse multiquadrics RBF (IMQ-RBF)

$$h^j(r, b^j) = h^j(\|\mathbf{x} - \mathbf{c}^j\|, b^j) = \frac{1}{\sqrt{r^2 + (b^j)^2}}, \tag{2.16}$$

- Thin plate splines RBF (TPS-RBF)

$$h^j(r) = h^j(\|\mathbf{x} - \mathbf{c}^j\|) = r^{2s} \log(r), \qquad s = 1, 2, ..., \tag{2.17}$$

- Compact support RBF (CS-RBF)

$$h^j(r) = h^j(\|\mathbf{x} - \mathbf{c}^j\|) = (1 - r)^l_+ p(r), \qquad l = 1, 2, ..., \tag{2.18}$$

where s is the order of TPS-RBF; $p(r)$ is a polynomial of Wendland CS-RBF (Wendland, 1995,1998). While the MQ-RBF, TPS-RBF and CS-RBF exhibit global response, i.e. they increase monotonically with increasing distance from the centre, the Gaussian-RBF and IMQ-RBF have local response (localized function) (Haykin, 1999; Beatson and Light, 1997). RBFs are multivariate as function of $\mathbf{x} \in \mathbf{R}^d$, but univariate as function of r, which should be a tremendous computational advantage if the space dimension is large.

2.2.3 RBFN for the approximation of a given function

In this work, we focus on the problem of estimating an unknown mapping $f : \mathbf{R}^d \to \mathbf{R}$ from a finite set S_n of n examples in which only the centres and linear weights need to be determined while kernel functions and other parameters are fixed beforehand. This process is considered as a supervised training of the networks with the training set S_n. The training set contains elements which consist of paired values of the independent (input) and dependent (output) variables and is represented by

$$S_n = \{(\mathbf{x}_i, \hat{y}_i)\}_{i=1}^{n}, \tag{2.19}$$

where $\mathbf{x}_i$ and $\hat{y}_i$, $i = 1, ..n$ are values of the training input and output vectors of the networks, respectively. Note that the output is assumed to be one-dimensional but this assumption does not in any way limit the general applicability of the network training being developed in this work. The hat over the symbol y_i shows that the output values of the training set are usually assumed to be corrupted by noise. In other words, the training set only specifies $\hat{y}_i$ which is equal to y_i (the correct output value to pair with $\mathbf{x}_i$) plus a small amount of unknown noise. In the following, the network training is described with the two cases: (i) the noise-free case ($\hat{y}_i = y_i$) and (ii) the noisy case ($\hat{y}_i = y_i + \epsilon_i$) where ϵ_i is a random process of the training set.

Approximation problem in the noise-free case

Let $\mathbf{c}_{i=1}^{m}$ be the centre vectors which may be chosen either as a subset of the training input vector ($m <= n$), or otherwise via a self-organizing procedure such as k-means clustering (Duda and Hart, 1973; Darken and Moody, 1990).

With the training set $S_n = \{\mathbf{x}_i, \hat{y}_i = y_i\}_{i=1}^{n}$, where $\{\hat{y}_i\}_{i=1}^{n}$ is the set of desired output values corresponding to $\{\mathbf{x}_i\}_{i=1}^{n}$, the weight vector of networks can be determined via the minimization of the following cost function

$$\mathcal{C}(\mathbf{w}) = \sum_{i=1}^{n} \left(\hat{y}_i - f(\mathbf{x}_i)\right)^2 . \tag{2.20}$$

Applying the linear least-squares principle leads to a system of linear algebraic equations of m unknown weights as follows (see Appendix B.2 for more detail)

$$(\mathbf{H}^{\mathbf{T}}\mathbf{H})\mathbf{w} = \mathbf{H}^{T}\hat{\mathbf{y}}, \tag{2.21}$$

where $\mathbf{H}$ is the $m \times n$ design matrix (or interpolation matrix) with $H_{ij} = h^{j}(x_i)$; m is the number of RBF-neurons and n is the number of training points, and $\hat{\mathbf{y}} = [\hat{y}_1 \quad \hat{y}_2 ... \hat{y}_n]^T$ is the n-dimensional vector of training output values. In the case where the centres are chosen to be identical to the training input vectors ($n = m$), (strict interpolation in multidimentional space), Eq (2.21) is rewritten as follows

$$\mathbf{H}\mathbf{w} = \hat{\mathbf{y}}. \tag{2.22}$$

The regularization approach to the approximation problem

The strict interpolation procedure with the least-squares principle in Eq (2.20) may not be a good strategy for RBFN training in the case where the networks may end up fitting misleading variation. It is due to idiosyncrasies or noise in the input data caused by the number of data points of S_n which is much larger than the number of degrees of freedom of the underlying mapping (Broomhead and Lowe, 1988 and Haykin, 1999). In other words, the reasons could be that (i) there may not be as much information in the training sample as really needed to reconstruct the input-output mapping uniquely; (ii) there is the presence of noise or imprecision in the training data; (iii) there is a distinct output which may not exist for every input. The learning is an ill-posed inverse problem which could be regularized with a suitable mechanism that allows the fit to deviate from the training data. Such a regularized least-squares solution essentially imposes a smoothness constraint on the estimate. The regularization approach determines a solution f of the approximation problem by minimizing the following functional (Girosi, 1992 and Girosi *et al.*, 1995)

$$\sum_{i=1}^{n} \left(\hat{y}_i - f(\mathbf{x}_i)\right)^2 + \lambda\phi(f), \tag{2.23}$$

where $\phi(f)$ is a smoothness functional and also called the stabilizer which enforces some smoothness constraint and needs to be chosen from a suitable class of stabilizers (see Haykin, 1999 for more details); λ is a positive parameter that is usually called the regularization parameter and controls the tradeoff between the two terms in (2.23). There are several model selection criteria for choosing a value of λ such as Bayesian information criterion, cross-validation (Wahba, 1990), General cross-validation (GCV) (Haykin, 1999), structural risk minimization (Vapnik, 1998) or some other principles which can be found in Haykin (1999) and Orr (1999a,b). The GCV is used in this work. The structure of $\phi(f)$ embodies the prior knowledge about the solution and the characteristic of the problem. If $\lambda = 0$, we return to the noise-free case while for $\lambda \to \infty$, it is the case of a low pass filter.

2.2.4 RBFN for the approximation of the solution of a PDE

During the last decade, many scientists and engineers have discovered the merits of RBFNs when working with multidimensional data in which the applications seem to have shifted from scattered data approximation to the numerical solution of partial differential equations. The variables of the governing PDEs are represented by networks which are designed simultaneously to satisfy the PDEs together with the boundary conditions. Two important features of RBFN-based method in solving PDEs are (i) it does not require any element (truly meshless method); (ii) it is space dimension independent in the sense that convergence order is of $O(\vartheta^{d+1})$ where d is the spatial dimension and ϑ is the density of collocation points (Wu, 1983 and Kansa and Hon, 2000).

The RBFN-based methods for numerical solution of PDEs can also be classified into several categories as follows.

- The particular solution method. RBFNs are used to find a particular solution (PS) of the inhomogeneous PDE. The general solution is the sum of the homogeneous fundamental solution and the PS (Golberg and Chen, 1997 and Tran-Canh and Tran-Cong, 2002a);
- The Meshless Galerkin method using RBFNs (Wendland, 1999). In this category, the unknowns are approximated by RBFs and the error functions (see §1.4) are distributed on the domain via the Galerkin method;
- The RBFN collocation method (Kansa's method). The variables in this method are approximated by RBFNs and then the distribution of the error functions is carried out on the domain under consideration via the collocation method. Both the PDEs and boundary conditions are satisfied by collocation (Kansa, 1990a,b; Fasshauer, 1996,1999; Franke and Schaback, 1998).
- The RBFN least-squares method. The variables are approximated by RBFNs, the least-squares scheme, instead of collocation method, is employed to distribute the errors on the analysis domain (Mai-Duy and Tran-Cong, 2001 and Tran-canh and Tran-Cong, 2002a,b).

  In approximating functions using RBFNs, there are two different directions: (i) the closed-form RBFN approximating a function (or unknown) is obtained from the set of training points and then their derivatives are determined directly by differentiating this RBFN (Kansa, 1990b, Zerroukat *et al.*, 1998; Tran-Canh and Tran-Cong, 2002a,b); or (ii) the formulation of the problem starts with the expression of the highest order derivatives of the unknown (existing in the PDEs) in terms of RBFNs and then the derivative expressions are integrated to obtain the approximating expression for the unknowns (Mai-Duy and Tran-Cong, 2001). Since its introduction (Kansa, 1990b), there have been several modifications to and improvements on the RBFN-based method, which can be summarized as follows.

- ◇ The straightforward RBF collocation method is proposed by Kansa (1990a,b) in which collocation with the boundary points satisfies the boundary conditions, and collocation with the interior points satisfies the PDEs;
- ◇ In the direct collocation, the PDE is collocated both on the boundary and the internal region (Fedoseyev *et al.*, 2002). The method reduces the error near boundaries in RBF-based approximation (Fornberg *et al.*, 2002);
- ◇ The symmetric collocation scheme modifies the basis functions in the approximant according to specific properties of the operator in the PDEs. This modification of RBF leads to a symmetric equation system (see Fasshauer (1996) for more details).

Depending on the problem, the modification used can be better in some cases but not in others. However, in this work, the second one is applied in solving PDEs for its error reduction capability. In the next chapters, this scheme will be described in more detail for individual applications as appropriate. Besides the advantages, the primary drawback of the RBFN-based meshless methods is that the coefficient matrices obtained from the global discretization scheme are full. This makes these matrices become more ill-conditioned as the rank increases. Several techniques have been proposed to improve the conditioning of the coefficient matrix and the solution accuracy, including (i) domain decomposition methods for large problems; (ii) knots adaptivity that minimizes the total number of knots required in a simulated problem; (iii) matrix preconditioners; (iv) replacement of global solvers by block partitioning etc. A combination of the approaches can contribute to more accurate solution. In this dissertation, the research focuses on the element-free RBFN-based least-squares method in conjunction with the domain decomposition for the numerical solution of PDEs.

2.3 The stochastic differential equation

Recently, the theory of SDEs has had application in a large number of disciplines such as physics, engineering, biology, communication, economics (Kubo *et al.*, 1978; van Kampen, 1981; Gardiner, 1994 and Kloeden and Platen, 1995), and especially in rheology where the basic equations describing the polymer dynamics, using the coarse-grained models of kinetic theory, are stochastic in nature (Öttinger, 1996). In order to describe a stochastic process, two of the most commonly used interpretations of the equations are the Ito and Strastonovich interpretations. The first one is most practical for computation and the second one appeals more to physical intuition. In this section, a brief review of the SDEs of an Ito process is presented, the other extension of the Ito SDEs relating to the present work will be introduced later in relevant chapters.

2.3.1 Introduction

Consider the multi-component SDE of an Ito process as follows

$$d\mathbf{Q}(t) = \mathbf{A}(t, \mathbf{Q}(t))dt + \mathbf{B}(t, \mathbf{Q}(t))d\mathbf{W}(t), \tag{2.24}$$

where $\mathbf{A}$ is a d-dimensional column vector and $\mathbf{A}(t,\mathbf{Q}(t))dt$ is referred to as the drift term; $\mathbf{B}$ is a $d \times d'$ matrix and $\mathbf{B}(t,\mathbf{Q}(t))d\mathbf{W}(t)$ is referred to as the diffusion term; $\mathbf{W}(t)$ is the Wiener process (Appendix C.3). In the present work, $\mathbf{Q}(t)$'s represent the configuration fields of polymer solution. The solution $\mathbf{Q}$ and the Wiener process $\mathbf{W}(t)$ are defined on the same probability space. More details on this problem can be found in Gihman and Skorohod (1974); Friedman (1975) and Garnier (1994). If $\mathbf{B} = \mathbf{0}$, (2.24) becomes a deterministic differential equation. Depending on whether $\mathbf{B}$ is a function of $\mathbf{Q}(t)$ or not, we have a multiplicative or additive noise process. In this dissertation, it is assumed that the SDE (2.24) satisfies the conditions for the existence and uniqueness of the solution. This assumption is considered to be accepted in many important classes of SDE used in the kinetic theory application (Öttinger, 1996).

2.3.2 Numerical integration schemes

Since the SDEs associated with non-linear models of polymer kinetic theory usually cannot be solved analytically, many numerical integration approaches have been developed. In this section, several numerical stochastic time discretisation methods employed in the Brownian simulation techniques in the present work are presented. In these methods the time evolution of individual realizations of a stochastic variable is simulated. Averaged quantities can be determined at any instant in time by statistically averaging over a large number of independent realizations. A detailed and comprehensive description of numerical schemes on SDEs can be found in Kloeden and Platen (1995), Öttinger (1996) and Kloeden *et al.* (1997).

□ The stochastic theta method (STM)

Considering the numerical solution of an Ito stochastic process given by (2.24)

$$d\mathbf{Q}(t) = \mathbf{A}(t, \mathbf{Q}(t))dt + \mathbf{B}(t, \mathbf{Q}(t))d\mathbf{W}(t). \tag{2.25}$$

The initial condition $\mathbf{Q}(t_o) = \mathbf{Q}_o$ is independent of the Wiener process $\mathbf{W}_t$. Applying the STM with fixed stepsize leads to the iterative series as follows

$$\mathbf{Q}_{i+1} = \mathbf{Q}_i + (1-\theta)\mathbf{A}(t_i, \mathbf{Q}_i)\Delta t + \theta \mathbf{A}(t_{i+1}, \mathbf{Q}_{i+1})\Delta t + \mathbf{B}(t_i, \mathbf{Q}_i)\Delta \mathbf{W}_i, \tag{2.26}$$

with

$$\begin{aligned} E(\Delta \mathbf{W}_i) &= 0, \\ E((\Delta \mathbf{W}_i)^2) &= \Delta t, \end{aligned} \tag{2.27}$$

where $t_i = i\Delta t$, $(i = 0, 1, \cdots, n-1)$, $\mathbf{Q}_i = \mathbf{Q}(t_i)$; $\Delta \mathbf{W}_i = \mathbf{W}_{i+1} - \mathbf{W}_i$ represents the $N(0; \Delta t)$ distributed increment of the standard Wiener process $\mathbf{W}$, θ $(\theta \in [0, 1])$ is a parameter which shows the degree of implicitness.

- For $\theta = 0$, we have the explicit Euler-Maruyama method and (2.26) is rewritten as follows

$$\mathbf{Q}_{i+1} = \mathbf{Q}_i + \mathbf{A}(t_i, \mathbf{Q}_i)\Delta t + \mathbf{B}(t_i, \mathbf{Q}_i)\Delta \mathbf{W}_i, \tag{2.28}$$

- For $\theta > 0$, the method is implicit. Especially with $\theta = 1$, the fully implicit Euler method is given from (2.26) by

$$\mathbf{Q}_{i+1} = \mathbf{Q}_i + \mathbf{A}(t_{i+1}, \mathbf{Q}_{i+1})\Delta t + \mathbf{B}(t_i, \mathbf{Q}_i)\Delta \mathbf{W}_i, \tag{2.29}$$

The implicit method is applied only in the drift term but not in the diffusion term. The increment $\Delta \mathbf{W}_i$ can be generated by the Box-Muller or Polar Marsaglia generator (Appendix C.4), or other random number generators. The Euler method can be interpreted as an order 0.5 strong Ito-Taylor approximation and is known to converge with order 1 for deterministic differential equations (Kloeden and Platen, 1995). The other methods with higher order strong Taylor approximation are not mentioned here and can be found in Kloeden and Platen (1995); Öttinger (1996).

□ **Predictor-Corrector methods (PCM)**

Predictor-Corrector methods are employed because of their numerical stability, which they inherit from the implicit counterparts of their corrector schemes. In this section, only the family of weak predictor-corrector methods is presented. The methods consist of two steps

- The predictor step uses the explicit schemes given by

$$\mathbf{Q}^*_{i+1} = \mathbf{Q}_i + \mathbf{A}(t_i, \mathbf{Q}_i)(\Delta t_i) + \mathbf{B}(t_i, \mathbf{Q}_i)(\Delta \mathbf{W}_i); \qquad (2.30)$$

- The corrector step uses implicit schemes, in which the predicted value $\mathbf{Q}^*_{i+1}$ is employed instead of $\mathbf{Q}_{i+1}$ on the RHS of the corrector step. The corrected value is expressed as follows

$$\mathbf{Q}_{i+1} = \mathbf{Q}_i + \left\{\theta \mathbf{A}(\mathbf{Q}^*_{i+1}, t_{i+1}) + (1-\theta)\mathbf{A}(\mathbf{Q}_i, t_i)\right\} \Delta t_i + \\ + \left\{\theta' \mathbf{B}(\mathbf{Q}^*_{i+1}, t_{i+1}) + (1-\theta')\mathbf{B}(\mathbf{Q}_i, t_i)\right\} \Delta \mathbf{W}_i, \quad (2.31)$$

where $\theta, \theta' \in [0, 1]$. The corrector (2.31) includes some degree of implicitness in the diffusion term ($\theta' > 0$). In predictor-corrector methods, the difference between the predicted and corrected values provides an indication of local errors at each step.

2.4 Brownian dynamics simulations (BDS) in polymeric kinetic theory

As described in chapter 1, in the micro-macroscopic modelling of viscoelastic flows, the continuity and momentum equations are considered together with the extra-stress derived from BDS. The development of the kinetic theory for polymers allows a hierarchical description of the fluid microstructure that can

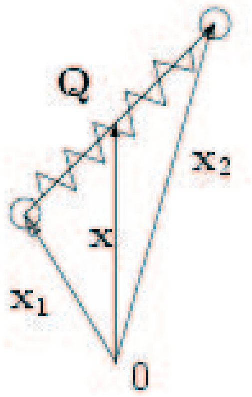


Figure 2.2: Elastic dumbbell model: the connector vector **Q** describes the configuration of the model.

be systematically reduced in dimensionality (coarse-grained). The approach, which is based on the dynamics of the polymer chains and their interaction with the solvent, gives rise to macro-molecular kinetic models. In these models, the polymer chains are modelled not at molecular level but as macroscopic objects with respect to the size scale of molecules. For example, a number of molecules are approximated as a chain of dumbbells. A dumbbell consists of a massless spring connecting two equal point masses which are related to the actual molecular mass. The coarsest approximation used in modelling a polymer chain is a single dumbbell. In this section, the BDS will be presented for simulating dilute polymer solutions using the elastic dumbbell.

2.4.1 Equation of motion of a dumbbell

A dumbbell can be described at any time t by either the coordinates of its beads $\mathbf{x}_i(t)$ $(i = 1, 2)$, or the coordinate of dumbbell's center of mass $\mathbf{x}(t)$ and the connector vector $\mathbf{Q}(t)$ (Fig. 2.2) as follows

$$\begin{aligned} \mathbf{x}(t) &= \frac{\mathbf{x}_1(t)+\mathbf{x}_2(t)}{2}, \\ \mathbf{Q}(t) &= \mathbf{x}_2(t)-\mathbf{x}_1(t). \end{aligned} \tag{2.32}$$

$\mathbf{Q}(t)$'s, also named configuration vectors, describe the orientation and elongation of polymer molecules and define a configuration space of the flow field of the polymer solution.

Neglecting the inertia term, other external forces and assuming that the masses of the beads as well as their acceleration are very small, the Newton's law for a dumbbell at a time t is given by

$$\zeta\left(\mathbf{u}(\mathbf{x}_i,t)-\frac{d\mathbf{x}_i(t)}{dt}\right)+\mathbf{F}^S(\mathbf{x}_{i'}(t)-\mathbf{x}_i(t))+\mathbf{F}_i^B(t)=0, \quad i=1,2, \qquad (2.33)$$

where i, i' are the indices of the two beads of a dumbbell (if $i=1$ then $i'=2$ and vice versa); ζ is the bead friction coefficient; $\mathbf{u}(\mathbf{x}_i,t)$ is the velocity of the solution at the considered bead at time t. In Eq (2.33), the forces acting on each bead include

- The hydro-dynamic drag force $(\mathbf{F}_i^D=\zeta\left(\mathbf{u}(t,\mathbf{x}_i(t))-\frac{d\mathbf{x}_i}{dt}(t)\right), i=1,2)$. This force is due to the average hydrodynamic resistance of the motion of the polymer through a viscous solvent. It is proportional to the difference between the bead velocity and the average velocity of the solution and tends to make the beads follow the flow;

- The Brownian force $(\mathbf{F}_i^B,\ i=1,2)$. It is due to the thermal fluctuation in the solution and the cumulative effect of the bombardment of the dumbbell by the solvent molecules. The average of this random force can be determined as an expression with respect to the configuration distribution function. The force is from the Brownian motion theory which can be found in Gardiner (1995) or Öttinger (1996), for example.

- The intramolecular (spring potential) force $(\mathbf{F}_i^S)$. This force comes from the spring of the dumbbell. The spring potential force is calculated according to the dumbbell force law $\mathbf{F}(\mathbf{Q})$ associated with particular molecular models which can be found in Bird *et al.* (1987b). In the present work, several elastic dumbbell models will be employed and presented in the relevant chapters.

The Brownian dynamics simulation based on the dynamics of the dumbbells involves the stochastic differential equations, which describe the internal configurations of the dumbbells and the expression of the extra-stress, which is related to the configurations of the dumbbells via a formula involving the expectation. In this section, the Brownian dynamics simulation is presented according to both the Lagrangian and Eulerian points of view.

2.4.2 Lagrangian Brownian dynamics simulation

Governing equation for a dumbbell

From the Brownian motion theory, the Brownian force is given by (chapter 3, Öttinger, 1996)

$$F^B(t) = \sqrt{2k_BT\zeta}\frac{d}{dt}\mathbf{W}(t), \tag{2.34}$$

where the factor $\sqrt{2k_BT\zeta}$ is dictated by fluctuation-dissipation theorem of the second kind, with k_B is the Boltzmann's constant; T is the absolute temperature and $\mathbf{W}(t)$ is a standard Wiener process. Integration of the Eq (2.33) for the beads of a dumbbell over $[0,t]$, making use of the Brownian force (2.34) and the Taylor expansion for velocity field, yields the integral form of the equations of motion of a dumbbell as follows.

$$\begin{aligned}
\mathbf{x}\,(t) &= \mathbf{x}(0) + \int_0^t \mathbf{u}\,(\mathbf{x}(t'),t')\,dt', && (2.35)\\
\mathbf{Q}\,(t) &= \mathbf{Q}\,(0) + \int_0^t \left(\boldsymbol{\kappa}\,(\mathbf{x}(t'),t')\cdot\mathbf{Q}\,(t') + \frac{2}{\zeta}\mathbf{F}\,(\mathbf{Q}\,(t'))\right)dt' \\
&\quad + \sqrt{\frac{4k_BT}{\zeta}}\int_0^t d\mathbf{W}\,(t'). && (2.36)
\end{aligned}$$

Equations (2.35) and (2.36) are often expressed in the form of stochastic differential equations as follows.

$$\begin{aligned}
d\mathbf{x}(t) &= \mathbf{u}(\mathbf{x}(t),t)dt, && (2.37)\\
d\mathbf{Q}(t) &= \left(\boldsymbol{\kappa}(\mathbf{x}(t),t)\cdot\mathbf{Q}(t) + \frac{2}{\zeta}\mathbf{F}(\mathbf{Q}(t))\right)dt + \sqrt{\frac{4k_BT}{\zeta}}d\mathbf{W}(t), && (2.38)
\end{aligned}$$

where $\boldsymbol{\kappa}$ is the velocity gradient and $\mathbf{u}(\mathbf{x}(t),t) = \mathbf{u}_o(\mathbf{x}) + \boldsymbol{\kappa}(\mathbf{x}(t),t) \cdot \mathbf{x}(t)$.

Equations (2.35)-(2.36) or (2.37)-(2.38) describe the evolution of the configuration of a dumbbell expressed by $(\mathbf{x}(t), \mathbf{Q}(t))$ from the Lagrangian point of view. In other words, $\mathbf{Q}(t)$ represents the elongation vector of a dumbbell moving along its trajectory. Although $\mathbf{x}(t)$ in (2.35) or (2.37) is not a proper stochastic process, it can be considered as a random variable because the initial position of the dumbbell is distributed according to a uniform law (Öttinger, 1996). Thus, at any time $t' \in [0,t]$, the configuration of a dumbbell $(\mathbf{x}(t), \mathbf{Q}(t))$ can be considered as a multivariate random variable.

The extra stress tensor formulation

The extra polymer stress $\boldsymbol{\tau}^p$ of a polymer solution as described in chapter 1 is obtained from the dynamics of polymer molecules. This relation between the polymer stress tensor (macroscopic property) and the configurations of the dumbbells $\mathbf{Q}'s$ of the dumbbells (microscopic quantity) has been established through various different works such as Giesekus, Kramers, Kramers-Kirkwood and Modified-Kramers (Bird *et al.* 1987b). In this work, the following Kramer formulation is employed to express the polymer contribution to the stress at a location $\bar{\mathbf{x}} = \mathbf{x}(t)$ and time t as follows.

$$\boldsymbol{\tau}^p(t,\bar{\mathbf{x}}) = n_d k_B T \left(E\Big(\mathbf{Q}(t) \cdot \mathbf{F}(\mathbf{Q}(t))|_{\mathbf{x}(t)=\bar{\mathbf{x}}} \Big) - I \right), \tag{2.39}$$

where n_d is the volume density of dumbbells in the solution; I is the unity tensor; $E\Big(\mathbf{Q}(t) \cdot \mathbf{F}(\mathbf{Q}(t))|_{\mathbf{x}(t)=\bar{\mathbf{x}}}\Big)$ denotes the expectation of $(\mathbf{Q}(t) \cdot \mathbf{F}(\mathbf{Q}(t)))$ of the dumbbells at time t having coordinates $\mathbf{x}(t) = \bar{\mathbf{x}}$.

2.4.3 Eulerian Brownian dynamics simulation

In the Lagrangian Brownian dynamics simulation, $\mathbf{x}(t)$ in the equations of motion (2.37)-(2.38) is nothing but the trajectory of the dumbbells. These governing equations for the connector (elongation) vector can be rewritten under the Eulerian point of view and Hulsen *et al.* (1997) proposed it as the Brownian configuration fields (BCF) method. In this approach, instead of writing an independent equation for the center of mass of a dumbbell, the configuration of the dumbbell is considered as a field variable $\hat{\mathbf{Q}}(\mathbf{x}, t)$. The equation of motion is rewritten as follows.

$$d\hat{\mathbf{Q}}(\mathbf{x}, t) = \left(-\mathbf{u}(\mathbf{x}, t) \cdot \boldsymbol{\nabla} \hat{\mathbf{Q}}(\mathbf{x}, t) + \boldsymbol{\kappa}(\mathbf{x}, t) \cdot \hat{\mathbf{Q}}(\mathbf{x}, t) - \frac{2}{\zeta} \mathbf{F}(\hat{\mathbf{Q}}(t)) \right) dt + \sqrt{\frac{4k_B T}{\zeta}} d\mathbf{W}(t), \quad (2.40)$$

for the differential form and

$$\hat{\mathbf{Q}}(\mathbf{x}, t) = \hat{\mathbf{Q}}(\mathbf{x}, 0) + \int_0^t \left(-\mathbf{u}(\mathbf{x}, t') \cdot \boldsymbol{\nabla} \hat{\mathbf{Q}}(\mathbf{x}, t') + \boldsymbol{\kappa}(\mathbf{x}, t) \cdot \hat{\mathbf{Q}}(\mathbf{x}, t') - \frac{2}{\zeta} \mathbf{F}(\hat{\mathbf{Q}}(\mathbf{x}, t')) \right) dt' + \sqrt{\frac{4k_B T}{\zeta}} \int_0^t d\mathbf{W}(t'), \quad (2.41)$$

for the integral form. The field variable $\hat{\mathbf{Q}}(\mathbf{x}, t)$ is a stochastic process and depends on $\mathbf{x}$ and t. Clearly, in these equations, the term $\mathbf{u}(\mathbf{x}, t) \cdot \boldsymbol{\nabla} \hat{\mathbf{Q}}(\mathbf{x}, t)$ accounts for the displacement of dumbbells along the fluid trajectories. The BCF method avoids the extra effort associated with a necessary particle tracking procedure.

In the Eulerian Brownian dynamics simulation, the extra polymer stress tensor is given by

$$\boldsymbol{\tau}^p(t, \mathbf{x}) = n_d k_B T \left(E\left(\hat{\mathbf{Q}}(\mathbf{x}, t) \cdot \mathbf{F}(\hat{\mathbf{Q}}(\mathbf{x}, t)) \right) - I \right). \quad (2.42)$$

In spite of the difference of the meaning of $\mathbf{Q}$ and $\hat{\mathbf{Q}}$ as described, for the sake of brevity, $\mathbf{Q}$ is used for both.

2.4.4 Variance reduction methods

In Brownian dynamics simulation, a large number of independent realizations is simulated. In macromolecular approaches, the macro-quantities use to be determined as a function of the average values of stochastic quantities. Such an average is approximated by their arithmetic mean as follows

$$\mu = \frac{1}{n}\sum_{i=1}^{n} q_i, \tag{2.43}$$

where q_i is a stochastic quantity; $i = 1, 2, ..., n$; n is the number of realizations. Hence, there exists a deviation from the theoretical expectation value of q_i's, which can be measured by the following root-mean-square (RMS) deviation

$$\begin{aligned}\sqrt{E\left((\mu - E(q_i))^2\right)} &= \sqrt{E\left(\frac{1}{n^2}\sum_i q_i^2 + \frac{1}{n^2}\sum_i\sum_{j\neq i} q_i q_j\right) - E\left(q_i\right)^2} \\ &= \sqrt{\frac{E\left(q_i^2\right) - E\left(q_i\right)^2}{n}} \\ &= \sqrt{\frac{Var(q_i)}{n}}, \end{aligned}\tag{2.44}$$

where $E(.)$ means the expectation of (.). A reduction of this deviation can be achieved by (i) increasing the number of realizations (n) or (ii) reducing directly the variance. Since the increase of realizations entails a higher computation time, the variance reduction methods, which are based on the same ideas as Monte-Carlo scheme, are chosen to decrease the variance. In this section, two classes of variance reduction methods, namely importance sampling and control variate, are described.

- The importance sampling method. The main idea behind this method is to sample another probability density $\bar{p}(x)$ instead of $p(x)$. A correction factor $\frac{p(x)}{\bar{p}(x)}$ is introduced to assure that the same $E(q)$ is obtained

$$E(q) = \int q(x)p(x)dx = \int q(x)\frac{p(x)}{\bar{p}(x)}\bar{p}(x)dx = E\left(q\frac{p}{\bar{p}}\right)_{\bar{p}} \tag{2.45}$$

The new variance associated with density $\bar{p}(x)$ is $Var(q\frac{p}{\bar{p}})$. This variance is zero if the term $q\frac{p}{\bar{p}}$ is independent of x. The choice of $\bar{p}$ depends on the variable to be averaged and so $\bar{p} = \frac{q(x)p(x)}{E(q)}$ is used. The implementation of the method for the numerical solution of a SDE is quite complex and more details can be found in Melchior and Öttinger (1995, 1996) and Kloeden and Platen (1995).

- The control variate method. The method uses a control variate $\bar{q}$ which is correlated with a random variable q, to produce a better estimator of $E(q)$. While $E(q)$ is unknown and needs to be estimated, $E(\bar{q})$ can be calculated by a deterministic method. Let $\bar{\mu}$ be the estimator of $E(\bar{q})$. Instead of approximating $E(q)$ by μ; $(\mu - \bar{\mu} + E(\bar{q}))$ is used, i.e.

$$E(q) = \mu - \bar{\mu} + E(\bar{q}), \tag{2.46}$$

and its associated variance is $Var(q)+Var(\bar{q})-2Cov(\mu, \bar{\mu})$, where $Cov(\mu, \bar{\mu})$ is the covariance of q and $\bar{q}$ and shows how they are correlated. If each control variate $\bar{q}_i$ is strongly correlated with q_i, then $2Cov(\mu, \bar{\mu})$ is a large positive number and the value of $Var(q) + Var(\bar{q}) - 2Cov(\mu, \bar{\mu})$ is smaller than the value of $Var(q)$. Using the same random numbers for the realizations q_i and $\bar{q}_i$ of q and $\bar{q}$ ensures a strong correlation between them (Melchior and Öttinger, 1996; Bonvin and Picasso, 1999 and Kloeden and Platen, 1995).

In the present work, the control variate technique is employed in conjunction with the CVFNNSS from the Lagrangian and Eulerian points of view to reduce the variation of the macro-quantities.

2.5 Conclusion

Some brief reviews of the basic tools concerning the computational methods for the present work such as BIEs, FFNNs, SDEs and Brownian dynamics simulations, were presented in this chapter.

- The FFNNs and especially RBFNs are described at two levels:
  - The universal approximators for explicit functions and their derivatives. This capability of FFNNs allow efficient treatment of volume integrals. Hence, FFNNs are combined with other methods to solve boundary integral equations. The training procedure, in which the shape parameters of NNs are chosen in advance, is to determine a set of weights of the NNs based on the training data set by optimizing the cost function in the sense of general linear least-squares principle;
  - The meshless method for solving differential equations. This capability allows approximation of a PDE's solution via a training procedure where the information of the PDE and the boundary and/or initial conditions are employed as the training data for the networks design.

  Depending on the source, training data could be noise-free or randomly noisy and correspondingly specific networks training procedures are also presented. 'Mesh' convergence of FFNN-based methods is assured according to the Cover's theorem, which can be found in Haykin (1999) and is not repeated in this work. Among the various kinds of RBFNs, the MQ-RBFNs and TPS-RBFNs are chosen for the present work because of their known superior capability for approximation (Franke, 1982).

- The SDEs together with several time discretisation methods are presented as tools to deal with the Brownian dynamics simulations. These computational methods, which are the microscopic component of the hybrid simulation for the calculation of complex viscoelastic flows, are described from the Lagrangian and Eulerian points of view. In order to reduce statistical error in Brownian simulations, variance reduction schemes are also described.

As a first step in the coupling of the neural-like network-based mesh-free method and another numerical method in the macroscopic simulation approach, the BEM-FFNN-based numerical scheme for analysis of a non-Newtonian fluid, namely the GNF, will be presented in the next chapter.

Chapter 3

FFNN-based element-free scheme in BEM computation of GNF flows

This chapter presents a FFNN boundary-element-only scheme for analysis of GNF flows (Tran-Canh *et al.*, 2000a and Tran-Canh and Tran-Cong, 2000b, 2002a). The volume integral arising from the inhomogeneous PDEs, especially from nonlinear effects, is approximated via a particular solution (PS) technique. MLPNs and RBFNs are used for global approximation of field variables. Hence volume discretisation, associated with the volume integral, is not required, and in this way, BEM maintains the advantage of being a boundary-element-only method. The iterative numerical formulation is achieved by viewing the material as being composed of a Newtonian base (artificially assigned with a constant viscosity which may be different from sub-domain to sub-domain) and the remaining component which is accordingly defined from the original constitutive equation. This decoupling of the nonlinear effects allows a Picard-type iterative procedure to be employed by treating the nonlinear term as a known forcing function. However, convergence is sensitive to the estimate of this forcing function and an adaptive subregioning of the domain is adopted to control the accuracy of the estimate of this nonlinear term. The criterion for subregioning is that the velocity gradient should not vary significantly in each sub-domain. This strategy enables convergence of the present method (BEM-NN) at power-law index as low as 0.2 for the difficult power law fluid. The use of MLPNs (instead of single layer perceptrons) and RBFNs is another contributing factor to the improved convergence performance. The overall scheme is very suitable for coarse-grain parallelization as each sub-domain can be independently analyzed within an iteration. Furthermore, within each sub-domain process there are other parallelizable computations. The present method is verified with circular Couette and planar Poiseuille flows of the power-law, Carreau-Yasuda and Cross fluids.

3.1 Introduction

Many practical simulations of polymer flows, especially in complex geometries, still rely on the use of GNF models, due to the difficulties in using realistic but very complicated viscoelastic constitutive equations. The aim of this chapter is to report a contribution to further developments of a numerical approach based on integral equations (IEs), and global function approximation by NNs with particular application in the numerical analysis of GNF flows. By using certain particular solution techniques, the direct computation of the volume integrals is avoided and the IE formulation results in boundary-element-only methods (Coleman *et al.*, 1991; Zheng *et al.*, 1991a,b; Zheng and Phan-Thien, 1992). However, volume discretisation may still be needed for the purpose of piecewise approximation of functions in the style of finite element (FE) representation. The FE-type volume discretisation can be avoided with the use of methods of global function approximation such as NN methods (Nguyen-Thien and Tran-Cong, 2000) where only an unstructured distribution of data points is required. To overcome the lack of convergence, when nonlinearity is strong, the present approach adopts multi-layer perceptron networks, radial basis function networks, adaptive subregioning based on velocity gradients and a particular solution technique to enhance the IE formulation. Florez *et al.* (2000) devised a new multiple domain dual reciprocity method (MD-DRM) and also studied similar flow problems involving inelastic non-Newtonian fluids. These authors concluded that at low power law indices (strong nonlinearities) subregioning is necessary to obtain convergence. Here the MLPNs prove to be better than the single layer perceptron networks and the RBFNs are easier to train than the MLPNs in the present applications. Furthermore, a linear boundary element scheme is developed where the geometric corner problem is resolved in general and all singular integrals and some special non-singular integrals are calculated in closed form. The resultant BEM-NN method is a BE-only method without FE-type volume discretisation and is illustrated with the problems of circular Couette and planar Poiseuille flow of several GNFs. Although one might argue that subregioning is a form of FE discretisation, the two approaches differ in two important respects. Firstly, FEs require rigid connectivity to define a consistent topology while the present subregioning does not require any element along the interfaces. Secondly, the number of subregions is generally very small in comparison with the number of FEs (only two subregions are required in the present work and based on the coarse number of collocation points the equivalent number of quadrilateral FEs would be 180). The design of subregions is quite flexible, simple and should not cause any difficulty even in problems with moving boundary. This chapter is organized as follows: the governing equations for GNFs are reviewed in §3.2, followed by the BEM-NN formulation in §3.3. Numerical examples are then discussed in §3.4 with a brief conclusion in §3.5.

3.2 Governing equation for GNFs

From the description given in chapter 1, the system of mass and momentum conservation equations for the steady state, isothermal and creeping flow of an incompressible fluid is rewritten as follows (ignoring gravity)

$$\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}=0,\quad \mathbf{x}\in\Omega, \tag{3.1}$$

$$\boldsymbol{\nabla}\cdot\mathbf{u}=0,\quad \mathbf{x}\in\Omega, \tag{3.2}$$

with boundary conditions

$$\mathbf{u}=\mathbf{u}_0,\quad \mathbf{x}\in\partial\Omega_u, \tag{3.3}$$

$$\mathbf{t}=\mathbf{n}\cdot\boldsymbol{\sigma}=\mathbf{t}_0,\quad \mathbf{x}\in\partial\Omega_t, \tag{3.4}$$

where $\boldsymbol{\sigma}$ is the total stress tensor and $\mathbf{u}$ is the velocity vector; $\mathbf{n}$ is the outward unit normal to the boundary $\partial\Omega=\partial\Omega_t+\partial\Omega_u$ of the volume Ω; $\mathbf{t}$ is the traction vector. Considering the Newtonian fluid as a special case of a more general non-Newtonian fluid, the total stress tensor is decomposed into a Newtonian and a non-linear component as follows

$$\boldsymbol{\sigma}=-p\mathbf{I}+\boldsymbol{\tau}=(-p\mathbf{I}+2\eta_N\mathbf{D})+(\boldsymbol{\tau}-2\eta_N\mathbf{D})=\boldsymbol{\sigma}^N+\boldsymbol{\tau}^p, \tag{3.5}$$

where p is the hydrostatic pressure arising from the incompressibility constraint (3.2), $\mathbf{I}$ is the unit tensor, $\mathbf{D}$ is the rate-of-strain tensor, $\boldsymbol{\tau}$ is the extra stress tensor, η_N is a conveniently chosen 'Newtonian' viscosity, $\boldsymbol{\sigma}^N$ is an arbitrarily defined 'Newtonian component', $\boldsymbol{\tau}^p$ is the corresponding non-linear component consequently defined by

$$\boldsymbol{\tau}^p=\boldsymbol{\tau}-2\eta_N\mathbf{D}. \tag{3.6}$$

The extra stress $\boldsymbol{\tau}$ is governed by the constitutive equation for GNF as

$$\boldsymbol{\tau}=2\eta(\dot\gamma)\mathbf{D}, \tag{3.7}$$

where the non-Newtonian viscosity η is a function of the magnitude of the rate-of-strain tensor $\dot\gamma=\sqrt{2\mathrm{D}_{ij}\mathrm{D}_{ji}}$ (e.g. see Bird *et al* ., 1987a). In the present work, several well known models are considered. The power-law model is given by

$$\eta=m\dot\gamma^{n-1}, \tag{3.8}$$

where n (dimensionless) and m (with units of $Pa.s$) are two parameters. The Carreau-Yasuda model is given by

$$\frac{\eta - \eta_\infty}{\eta_o - \eta_\infty} = \left[1 + (\lambda\dot{\gamma})^a\right]^{\frac{n-1}{a}}, \tag{3.9}$$

where η_o is the zero-shear-rate viscosity, η_∞ is the infinite shear rate viscosity, λ is a time constant, n is the 'power-law exponent', a is a dimensionless parameter describing the transition region between the zero-shear-rate region and the power-law region. For many concentrated polymer solutions and melts, $a = 2$ and $\eta_\infty = 0$, and (3.9) is known as the Carreau model (Bird *et al* ., 1987a) in the form of

$$\eta = \eta_o \left[1 + (\lambda\dot{\gamma})^2\right]^{\frac{n-1}{2}}. \tag{3.10}$$

Finally, the Cross model is given by (Macosko, 1994)

$$\frac{\eta - \eta_\infty}{\eta_o - \eta_\infty} = \frac{1}{1 + (K^2\dot{\gamma}^2)^{\frac{1-n}{2}}}, \tag{3.11}$$

where K is a time constant and the other parameters are as defined above in the Carreau-Yasuda model.

3.3 BEM-PS-NN formulation for GNFs

3.3.1 PS and NN-based meshless method for the elimination of volume integral

For a given spatial point $\mathbf{x}$ of (3.1)-(3.4), noting (3.5), the integral representation (2.8) is rewritten as follows (see §2.1, chapter 2)

$$C_{ij}(\mathbf{x})u_j(\mathbf{x}) = \int_\Gamma u^*_{ij}(\mathbf{x},\mathbf{y})t_j(\mathbf{y})d\Gamma(\mathbf{y}) - \int_\Gamma t^*_{ij}(\mathbf{x},\mathbf{y})u_j(\mathbf{y})d\Gamma(\mathbf{y}) - \int_\Gamma \tau^p_{jk}n_k(\mathbf{y})u^*_{ij}(\mathbf{x},\mathbf{y})d\Gamma(\mathbf{y}) + u^p_i, \tag{3.12}$$

where $\mathbf{x}, \mathbf{y} \in \Omega$; $u_j(\mathbf{y})$ is the j-component of the velocity at $\mathbf{y}$; $t_j(\mathbf{y})$ is the j-component of the boundary traction at $\mathbf{y}$; $\tau^p_{jk}(\mathbf{y})$ is the jk-component of $\boldsymbol{\tau}^p$

at $\mathbf{y}$; $u^*_{ij}(\mathbf{x},\mathbf{y})$ is the i-component of the velocity at $\mathbf{x}$ due to a point force in j-direction at $\mathbf{y}$ (Kelvin fundamental solution corresponding to the Newtonian base material, see Appendix A.1) and $t^*_{ij}(\mathbf{x},\mathbf{y})$ is its associated traction. $C_{ij}(\mathbf{x})$ depends on local geometry and is described in §2.1. The volume integral u^p_i is defined by

$$\begin{aligned} u^p_i &= \int_\Omega u^*_{ij}(\mathbf{x},\mathbf{y})\frac{\partial\varepsilon_{jk}(\mathbf{x},\mathbf{y})}{\partial x_k}d\Omega(\mathbf{y}) \\ &= \int_\Omega G_{ij}(\mathbf{x},\mathbf{y})\phi_j(\mathbf{y})d\Omega(\mathbf{y}), \end{aligned} \tag{3.13}$$

where $\phi_j(\mathbf{y}) = \frac{1}{2}\frac{\partial\varepsilon_{jk}(\mathbf{x},\mathbf{y})}{\partial x_k}$, and $G_{ij}(\mathbf{x},\mathbf{y}) = 2u^*_{ij}(\mathbf{x},\mathbf{y})$ is twice the Kelvin fundamental solution.

The term

$$b_i(\mathbf{x}) = -\int_\Gamma \tau^p_{jk}n_k(\mathbf{y})u^*_{ij}(\mathbf{x},\mathbf{y})d\Gamma(\mathbf{y}) + u^p_i, \tag{3.14}$$

is treated as a pseudo body force which is estimated using the data obtained in the previous iteration or some initial guess for the first iteration. In the present work the Newtonian solution is used as the initial guess. Since the formulation of the problem is basically a Newtonian one with the non-Newtonian effects acting as a perturbation, it is natural to choose the Newtonian solution as an initial guess. Indeed, when the initial guess is set arbitrarily to zero on the interfaces betwen subregions, convergence is difficult to achieve. The compatibility conditions imposed on the primary variables (velocity and traction) ensure that convergence, if achieved, is global. The term (3.13) can be estimated without numerical volume integration by a particular solution technique (Nguyen-Thien *et al.*, 1997). Extra stress can also be calculated without volume discretisation by a neural network method similar to that used in Nguyen-Thien and Tran-Cong (2000) where single hidden layer perceptron neural networks were used. In order to improve convergence, different NN architectures are used in the present work as described in later sections. FFNN, as a universal global approximator or scattered data interpolator, only needs a random distribution of data points in the domain. In the present IE formulation, these data points are calculated point by point by a re-application of the IE for domain points. Since these data points are not simultaneous unknowns in the formulation they can easily be calculated in parallel. Once a NN approximation of the scattered data is found, the derivatives, which are needed in the calculation of the extra stress and pseudo body force, can be calculated in a straightforward manner owing to the closed form nature of the function decomposition into differentiable bases.

The boundary of the domain is discretised into a number of standard finite elements (i.e. boundary elements) and the discrete version of the IE (3.12) can now be written in the following matrix form

$$[\mathbf{G}]\,\{\mathbf{t}\} = [\mathbf{H}]\,\{\mathbf{u}\} + \{\mathbf{b}\}, \tag{3.15}$$

where $[\mathbf{G}]$ and $[\mathbf{H}]$ are the coefficient matrices resulting from the discretisation of the boundary into BEs; $\{\mathbf{t}\}$, $\{\mathbf{u}\}$ and $\{\mathbf{b}\}$ are the global nodal traction vector, velocity vector and pseudo body force vector, respectively. Hence, the problem is solved numerically by decoupling the nonlinear effects which are represented by $\{\mathbf{b}\}$ and treated as known small perturbations in an otherwise linear problem. The procedure is started with an initial guess of the solution, which is the Newtonian solution in the present application (i.e. $\{\mathbf{b}\} = \{0\}$). A Picard-type iteration is performed where the pseudo body force vector is estimated using the data obtained in the previous iteration until convergence or divergence is ascertained according to the following convergence measure

$$CM1 = \sqrt{\frac{\sum_1^N \sum_{i=1}^2 (u_i^n - u_i^{n-1})^2}{\sum_1^N \sum_{i=1}^2 (u_i^n)^2}} < tol, \tag{3.16}$$

where tol is a preset tolerance, u_i is the i component of the velocity at a node, N is the total number of nodes and n is the iteration number.

In summary, the computational steps are

(1.) If necessary, the analysis domain is divided into a small number of subdomains (*coarse-grain parallelisable*);

(2.) Specify the boundary conditions on the artificial interfaces according to the method described in Davies and Mushtaq (1996) (Newtonian values for the first iteration);

(3.) Then for each subdomain or the whole domain (the case of no domain decomposition)

(a.) the BE equations (3.15) are solved with the assumed pseudo body force (zero for the first iteration);

(b.) calculate the domain velocity field by a re-application of Eq. (3.12) point by point (*further parallelisable*);

(c.) approximate the continuous velocity field by an appropriate neural network. Each velocity component is represented by a different NN and therefore the process can be *further parallelised*;

(d.) calculate the extra stress field point by point (*further parallelisable*);

(e.) approximate the continuous extra stress field by an appropriate neural network. Each stress component is represented by a different NN and therefore the process can be *further parallelised*;

(f.) check for subdomain convergence (i.e. $CM1 < tol$ for each subdomain); go to 4 on convergence;

(g.) calculate the particular solution point by point (*further parallelisable*);

(h.) calculate the pseudo body force point by point (*further parallelisable*);

(i.) go to step 3a.

(4.) If there are subdomains, the 'master' program collects results from individual 'slave' programs and performs the book-keeping associated with the numerical domain decomposition scheme; check for convergence or divergence and go back to 2 or stop on convergence or divergence. The convergence measure for this step is defined as

$$CM2 = \sqrt{\frac{\sum_{interface=1}^{F} \sum_{i=1}^{N} (u_i^I - u_i^{II})^2}{\sum_{interface=1}^{F} \sum_{i=1}^{N} (u_i^{II})^2}}, \tag{3.17}$$

where $N = 2$ for 2D problems; I and II denote the two subregions sharing the $interface$ and F is the total number of interfaces;

Although the aim of this chapter is not a study in parallelization, the latter is simulated on a serial machine and indicated in the above algorithm for general interest. The results obtained here indicate that the MLPN procedure takes about 70 percent of the CPU time and therefore parallel computation is important in achieving high throughput. One way to achieve parallelization is to adopt the domain decomposition (DD) technique. Furthermore, it is found that the accuracy of the estimate of the pseudo body force has strong influence on whether the procedure is convergent and it is shown in a later §3.4 that DD can help achieve the required accuracy and improve convergence. The details of the steps in the above algorithm are discussed in the following sections.

3.3.2 Non-dimensionalisation

The velocity is scaled by U, the length dimension by a, the viscosity by η_*, the stress by $\eta_*\frac{U}{a}$ the dimensionless variables are given by

$$u' = \frac{u}{U}; l' = \frac{l}{a}; \eta' = \frac{\eta_n}{\eta_*}; \boldsymbol{\tau}' = \frac{\boldsymbol{\tau}}{\eta_*\frac{U}{a}}; \boldsymbol{\tau}^{p'} = \frac{\boldsymbol{\tau}^p}{\eta_*\frac{U}{a}}; \mathbf{D}' = \frac{\mathbf{D}}{\frac{U}{a}}, \tag{3.18}$$

where η_* varies from model to model.

For the power-law model, from (3.7) and (3.8), the extra stress is now written as

$$\boldsymbol{\tau} = 2m\dot{\gamma}^{n-1}\mathbf{D}, \tag{3.19}$$

and in dimensionless form

$$\boldsymbol{\tau}' = 2\frac{m}{\eta_N}\left(\frac{U}{a}\right)^{n-1}\dot{\gamma'}^{n-1}\mathbf{D}'. \tag{3.20}$$

Let $L = (\frac{m}{\eta_N})^{\frac{1}{n-1}}$ (the dimensional unit is second), where $\eta_* = \eta_N$, (3.20) can now be written as

$$\boldsymbol{\tau}' = 2\left(L\frac{U}{a}\right)^{n-1}\dot{\gamma'}^{n-1}\mathbf{D}' = 2\left(W\dot{\gamma'}\right)^{n-1}\mathbf{D}', \tag{3.21}$$

where the dimensionless group

$$W \equiv L\frac{U}{a} \tag{3.22}$$

is also called the Weissenberg number (Khellaf and Lauriat, 2000).

For the Carreau-Yasuda model, from (3.7) and (3.9), the extra stress is

$$\boldsymbol{\tau} = 2\left((\eta_o - \eta_\infty)\left[1 + (\lambda\dot{\gamma})^a\right]^{\frac{n-1}{a}} + \eta_\infty\right)\mathbf{D}, \tag{3.23}$$

and, similarly, the dimensionless form is

$$\boldsymbol{\tau}' = 2\left(\left[1 + W^a\dot{\gamma}'^a\right]^{\frac{n-1}{a}} + \tilde{\eta}\right)\mathbf{D}', \tag{3.24}$$

where $W = \lambda\frac{U}{a}$ is the Weissenberg number; $\eta_* = \eta_o - \eta_\infty$ and $\tilde{\eta} = \frac{\eta_\infty}{\eta_*}$. In the case where $a = 2$ and $\eta_\infty = 0$, the model reduces to the Carreau model

$$\boldsymbol{\tau}' = 2\left[1 + W^2\dot{\gamma}'^2\right]^{\frac{n-1}{2}}\mathbf{D}', \tag{3.25}$$

where $\eta_* = \eta_o$.

For the Cross model, from (3.7) and (3.11), the extra stress is

$$\boldsymbol{\tau} = 2\left((\eta_o - \eta_\infty)\left[\frac{1}{1 + (K\dot{\gamma})^{1-n}}\right] + \eta_\infty\right)\mathbf{D}, \tag{3.26}$$

and in dimensionless form

$$\boldsymbol{\tau}' = 2\left(\left[\frac{1}{1 + (W\dot{\gamma})^{1-n}}\right] + \tilde{\eta}\right)\mathbf{D}', \tag{3.27}$$

where $W = K\frac{U}{a}$ is the Weissenberg number; $\eta_* = \eta_o - \eta_\infty$; and $\tilde{\eta} = \frac{\eta_\infty}{\eta_*}$.

The IE (3.12) is now also in dimensionless form where the viscosity η_N in the relevant kernel is non-dimensionalised to $\eta' = \frac{\eta_N}{\eta_*}$ and η_* is dependent on particular models as mentioned above.

3.3.3 Boundary elements

In the present work, the 2D boundary elements used are straight line segment linear elements. The problem of multi-valued traction at a geometric corner is resolved in general and all singular integrals and some special case non-singular integrals are calculated analytically (see Appendix A.2) while other non-singular integrals are estimated using numerical quadrature.

3.3.4 Artificial neural networks (ANN)

The discrete velocity field obtained in the algorithm step 3b is to be approximated by a NN in the algorithm step 3c. Similarly, the discrete extra stress field obtained in the algorithm step 3d is to be approximated by a NN in the algorithm step 3e. A general discussion of NNs, including diagramatic representation of network architectures, was given in chapter 2 and can be found in many texts (Rojas, 1996; Mitchell, 1997; Haykin, 1999; Fine, 1999). Although FFNNs with only one hidden layer are capable of universal approximation, a practical rule for the determination of the number of neurons for a given accuracy is lacking. Furthermore, the corresponding 'training' to determine the synaptic weights is a difficult optimisation problem. In the present work it is found that it is difficult to achieve convergence for highly shear thinning fluids with single layer perceptrons. However, the single layer RBFNs as well as MLPNs are found to be suitable. Furthermore, the training sets in the present application are expected to be noisy due to numerical errors during iteration and therefore it is necessary to filter these noises in order to obtain the correct approximation (Rojas, 1996; Fine, 1999). For example, the sensitivity to peculiarities can be reduced by comparing models made up of different subsets of basis functions drawn from the same fixed set of candidates. Alternatively, regularisation can be employed, which is the case in the present work. Experience shows that training with regularisation (specially for MLPNs) yields better results. For example, the accuracy increases more than 8% in the function approximation and about 3% for its derivatives.

3.3.5 Multi-layer perceptron neural networks (MLPNs)

Approximation by MLPNs

As mentioned in chapter 2, a scalar function $f(\mathbf{x})$, $\mathbf{x} \in R^N$ can be represented by a MLPN architecture $(N - h_1....h_n - 1)$ where N is the number of neurons in the input layer corresponding to the dimension of the input data space, h_i is the number of neurons in the ith hidden layer of n hidden layers and the output layer has one neuron corresponding to the scalar function f. In the present work, MLPNs of two hidden layers are employed and the function is computed by the network according to (Fig. 3.1)

$$f(\mathbf{x}) = \sum_{k=1}^{m2} w_o^{1k} A_{h2}^k \left(\sum_{j=1}^{m1} w_{h2}^{kj} A_{h1}^j (\mathbf{x}^T \mathbf{w}_{h1}^j) \right), \tag{3.28}$$

where $m1$ is the number of neurons in the 1st hidden layer; $m2$ the number of neurons in the 2nd hidden layer; w_o^{1k} the synaptic weights associated with the connection between the output neuron (output layer) to the neuron k of the 2nd hidden layer; w_{h2}^{kj} the synaptic weights associated with the connection between the neuron k of the 2nd hidden layer to the neuron j of the 1st hidden layer; $\mathbf{w}_{h1}^j$ the vector of synaptic weights associated with the connection between the neuron j of the 1st hidden layer and the input layer; and $A_h^j(.)$ the chosen activation functions. The set of weights are determined by the training process. The log-sigmoidal function is here chosen (see §2.2.1, chapter 2) for the two hidden layers and the purelin function for the output layer. After the training process is successfully completed and the set of weights corresponding to the chosen architecture is obtained, the first order derivatives of the function are calculated by

$$\frac{\partial f(\mathbf{x})}{\partial x_i} = \sum_{k=1}^{m2} w_o^{1k} w_{h2}^{kj} DA \left(\sum_{j=1}^{m1} w_{h1}^{ji} DA(\mathbf{x}^T \mathbf{w}_{h1}^j) \right), \quad i = 1, \cdots, m, \tag{3.29}$$

where

$$DA(p) = \frac{dA(p)}{dp} = A(p)\,[1 - A(p)] \tag{3.30}$$

is the derivative of the log-sigmoidal activation function.

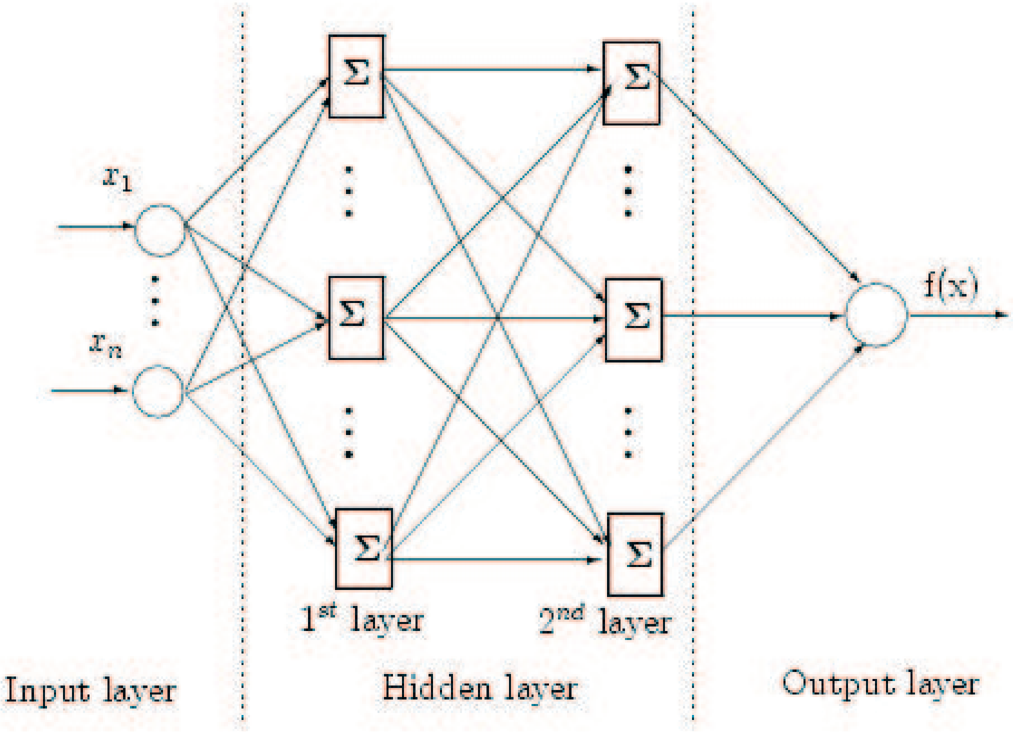


Figure 3.1: Architecture of a fully connected multi-layer perceptron network with two hidden layers.

Training MLPNs with backpropagation algorithm (BPA)

With a structure of a MLPN chosen and a given training set of n patterns $\{(\mathbf{x}_i, \hat{y}_i)\}_{i=1}^{n}$, the determination of a set of weights (unknowns) is based on the minimization of the following objective function (Moody and Darken, 1989; Michell, 1997)

$$S = \sum_{j=1}^{n} \left(\hat{y}_i - f(\mathbf{x}_i)\right)^2 + \lambda \left(\sum_{j,i} \left(w_{h1}^{ji}\right)^2 + \sum_{k,j} \left(w_{h2}^{kj}\right)^2 + \sum_{k} \left(w_{o}^{1k}\right)^2 \right), \quad (3.31)$$

where $f(\mathbf{x}_i)$ is given by (3.28) and λ is the global regularisation parameter. There is no ‘best’ algorithm for finding the weights of FFNNs in general, and MLPNs in particular (Fine, 1999). In the present work, Levenberg-Marquard method (see Appendix B.1) with error back-propagation is used because it is specially adapted to the minimization of the present objective function and the BPA organizes efficiently the gradient calculation for networks having more than one hidden layer (Michell, 1997; Demuth and Beale, 1998; Fine, 1999). There are several heuristics for the estimate of λ (Sen and Srivastava, 1997; Haykin, 1999; Orr, 1999a,b) of which the generalised cross-validation is used in the present work to optimize the regularisation parameter λ.

3.3.6 Radial basis function networks (RBFNs)

Approximation by RBFNs

A brief review of function approximation using the RBF network was presented in §2.2, chapter 2, and further details are given here in which the function $f(\mathbf{x})$ is decomposed into m fixed RBFs as (linear in $w^j, j = 1, \cdots, m$)

$$f(\mathbf{x}) = \sum_{j=1}^{m} w^j h^j(\mathbf{x}), \tag{3.32}$$

where w^j are synaptic weights and h^j are chosen radial basis functions. Let n be the number of input data points $\{\mathbf{x}_i, \hat{y}_i\}_{i=1}^{n}$, usually $m \leqslant n$. After the training process is completed and a set of weights corresponding to the chosen radial basis functions is obtained, the first order derivatives of $f(\mathbf{x})$ can be calculated as follows

$$\frac{\partial f(\mathbf{x})}{\partial x_i} = \sum_{j=1}^{m} w^j \frac{\partial h^j}{\partial x_i}. \tag{3.33}$$

The MQ-RBF and TPS-RBF are recognised mathematically for giving better approximation results by Franke (1982). However, the TPS-RBF has the advantage of containing no adjustable width parameter (see §2.2.2). Both the global MQ-RBF and the TPS-RBF are employed in this chapter. Specifically, the MQ-RBF and its derivatives are given by

$$h^j(r) = h^j(\|\mathbf{x} - \mathbf{c}^j\|) = \sqrt{r^2 + (b^j)^2}, \tag{3.34}$$

$$\frac{\partial h^j}{\partial x_i} = \frac{x_i - c_i^j}{\sqrt{r^2 + (b^j)^2}}, \tag{3.35}$$

where $\mathbf{r} = (\mathbf{x} - \mathbf{c}^j)$ and $r = \|(\mathbf{x} - \mathbf{c}^j)\|$ is the Euclidean norm of $\mathbf{r}$; $b^j > 0$ is the width of the jth RBF. The accuracy of the approximation can be very dependent on the width of the RBF (Kansa, 1990a; Carlson and Foley, 1991), of which the choice is still an open question. In the present work the approach proposed in (Carlson and Foley, 1991) is followed and the width is calculated by

$$b^j = kd^j, \tag{3.36}$$

where d^j is the distance from an RBF centre to its nearest neighbor and $k = 1.25$ is an empirical factor. The estimates of d^j were detailed in (Carlson and Foley, 1991).

In the case of the TPS-RBF, the function and its first order derivatives are given by

$$h^j(r) = h^j(\|\mathbf{x} - \mathbf{c}^j\|) = r^{2s}\log(r), \qquad s = 1, 2, ..., \tag{3.37}$$

$$\frac{\partial h^j}{\partial x_i} = r^{2(s-1)}(x_i - c_i^j)(2s\log r + 1). \tag{3.38}$$

Since the MQ-RBF is C^∞-continuous, it can be employed directly. In the case of TPS-RBF, it is C^{2s-1}-continuous, the power index s must be appropriately chosen for a given partial differential operator. In the present work, $s = 1$ is sufficient to satisfy the continuity condition. However, the case $s = 2$ is also investigated and the corresponding RBFNs are denoted by TPS-RBFN1 and TPS-RBFN2 respectively.

Training the linear RBFNs

The training of the model (3.32) given a training set of n patterns $\{(\mathbf{x}_i, \hat{y}_i)\}_{i=1}^n$, which is described in §2.2.3, chapter 2, can be achieved via the minimisation of a cost function based on the sum of squared errors. To counter the effects of over-fitting, a roughness penalty term can be added to the cost function to produce

$$C(\mathbf{w},\lambda) = \sum_{i=1}^{n} (\hat{y}_i - f(\mathbf{x}_i))^2 + \lambda \sum_{j=1}^{m} (w^j)^2, \tag{3.39}$$

where λ is the global regularisation parameter. Given a choice for λ, the minimisation of the cost function (3.39) yields an optimal weight vector $\mathbf{w} = [w^1, w^2, ..., w^m]^T$ according to

$$\mathbf{G}\mathbf{w} = \mathbf{H}^T\hat{\mathbf{y}}, \tag{3.40}$$

where $\mathbf{H}$ is the design matrix with $H_{ij} = h^j(x_i)$ and $\hat{\mathbf{y}} = [\hat{y}_1\hat{y}_2...\hat{y}_n]^T$ is the n-dimensional vector of training output values. $\mathbf{G} = \mathbf{H^T H} + \lambda\mathbf{I}$ is the variance matrix and $\mathbf{I}$ is the identity matrix (more detail can be found in Appendix B.3). There are several heuristics for the estimate of λ (Sen and Srivastava, 1997; Haykin, 1999) of which the generalised cross-validation is used in the present work to optimize the regularisation parameter λ.

Solving the least squares problems

One method for a least squares problem is to solve the corresponding normal equation in which the Gaussian elimination is employed to solve (3.40). However, this method may not be stable in the presence of round-off errors because of the high condition number of the normal matrix (Bjorck, 1996). The QR method, namely method of orthogonal triangle decomposition with pivoting, which can solve a wider class of least-squares problem for an over-determined linear system of equation (Bjock, 1996 and Lawson and Hanson, 1974), is used in this chapter.

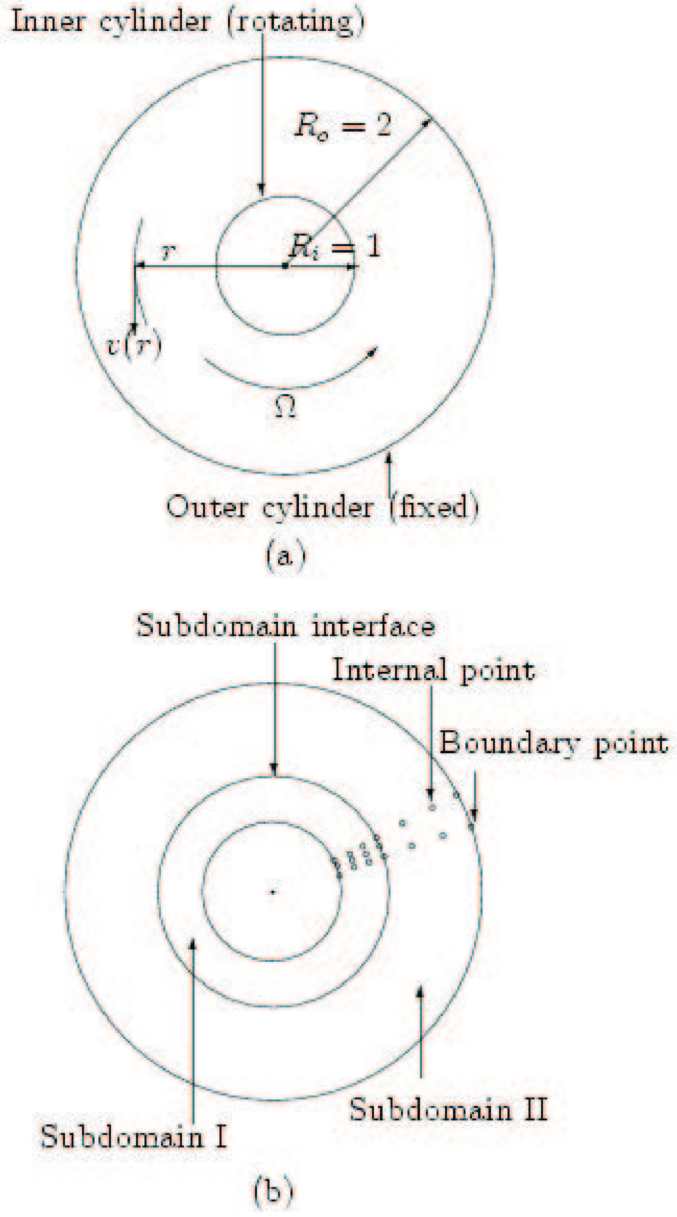


Figure 3.2: Circular Couette flow problem: (a) the outer cylinder is fixed and the inner concentric cylinder rotates with a constant speed Ω; non-slip boundary conditions at the walls are assumed (b) schematic subregioning of the flow domain where the point distribution is only schematic.

3.4 Numerical examples

The approach presented above is illustrated principally in this section with the analysis of the circular Couette flow and planar Poiseuille flow of three typical GNFs, described the power-law, the Carreau-Yasuda and the Cross models. The power-law fluid is particularly difficult to simulate at low power-law indices (high shear thinning) where it is helpful to employ the DD technique to alleviate the difficulty.

3.4.1 Circular Couette flow Problem

In this section, the circular Couette flow problem is considered where the fluid is confined between two infinitely long and concentric cylinders, the outer cylinder is fixed and the inner cylinder rotates at a constant angular speed (Fig. 3.2a). Let $R = \frac{R_o}{R_i}, \quad T_o = m\Omega^n R_i^2 = \eta_o \Omega R_i^2$, where R_o is the radius of the outer cylinder; R_i is the radius of the inner cylinder; Ω is the angular speed of the inner cylinder; T_o is the torque exerted on the outer cylinder. With regard to the non-dimensionalization scheme described in §3.3.2, $U = \Omega R_i, \quad a = R_i$.

Power-law model

The analytical solutions for the tangential velocity and the torque in dimensionless form are given by $(1 \leq r \leq R)$

$$v(r) = \frac{1}{R^{2/n} - 1}\left(\frac{R^{2/n} - r^{2/n}}{r^{(2-n)/n}}\right), \tag{3.41}$$

$$T = 2^{(n+1)}\pi R^2 \left(\frac{1}{n\left(R^{2/n} - 1\right)}\right)^n. \tag{3.42}$$

Numerical results are obtained for the following specific parameters: $R = 2$, the power-law index n is in the range of 0.2 to 0.7 and $W = 1$ (W is defined by §3.3.2).

Table 3.1: BEM-MLPN for circular Couette flow of power law fluids: maximum relative errors of the velocity field (single domain) in comparison with analytical results for different power law indices. N_b: number of boundary points, N_i: number of internal points, er: maximum relative error of the velocity field, n_i: number of iterations, $CM1$: convergence measure.

n	N_b	N_i	n_i	$CM1$	$er\%$
0.7	72	144	16	10^{-4}	0.97
0.6	72	144	19	10^{-4}	1.13
0.5	84	168	43	10^{-4}	1.65
0.4	84	168	61	10^{-4}	2.42

Results for the single domain case

This case is used to illustrate the parallelization of the computation related to the NN approximation of individual variables (velocity components and stress components). In other words, each scalar component is handled by a 'slave computing unit'. For $n = 0.7$ and $n = 0.6$, the number of boundary points and internal points is 72 and 144 respectively. For $n = 0.5$ and $n = 0.4$, the number of boundary points and internal points is 84 and 168 respectively. Table 3.1 shows the maximum of the errors of the velocity field with the values of power-law index n in the range of 0.7 to 0.4.

It can be seen from Table 3.1 that the solution accuracy deteriorates with decreasing power-law index n. This is expected as stronger non-linearity is associated with lower power-law indices n. High accuracy can be maintained in this case with "mesh refinement", i.e. with an increase in data density. However, increasing the data density adversely affects the convergence rate due to the optimization nature of MLPN training. Furthermore, for $n < 0.4$ convergence is not achieved. To overcome this problem the DD technique can be used as shown in the next section.

Results for the 2-subdomain case

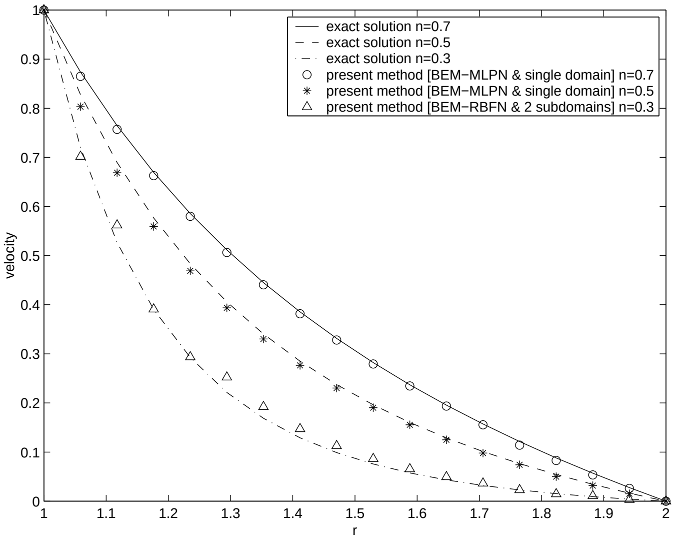


Figure 3.3: Circular Couette flow of power-law fluid: comparison of the present BEM-RBFN and BEM-MLPN results with the exact (analytical) solution where the tangential velocity profiles correspond to several values of the power-law index n.

The primary objective of DD here is to overcome the lack of convergence at low power-law indices (high non-linearity). As a beneficial consequence the DD process lends itself to coarse-grain parallelization. To verify the working of the DD scheme, the cases of $n = 0.7$, 0.5 and 0.4 are also analyzed and the results are compared with those obtained using a single domain. According to the AVGDD (adaptive velocity gradient domain decomposition) scheme (Tran-Canh *et al.*, 2000a,2002a), subregioning depends upon the variation of the velocity field. The non-linear component of the stress tensor (3.6) contains a velocity-gradient-dependent viscosity which is highly non-linear. This non-linear stress component contributes to the forcing term in the BIE formulation. By ensuring that the velocity gradient is similar in a domain of interest, the non-linear stress component remains relatively uniform in the domain and hence a more accurate NN approximation can be achieved for the same level of discretisation, which in turn enhances the convergence ability of the Picard-type iterative procedure used here. In the present circular Couette flow problem it is found that the velocity gradient is higher near the inner cylinder (Fig. 3.3). By

Table 3.2: BEM-MLPN for circular Couette flow of power-law fluids: maximum relative error of velocity field (two subdomains) in comparison with analytical results for different power law indices n. $r_i^I = R_i = 1.0$: internal radius of subdomain I, $r_o^{II} = R_o = 2.0$: external radius of subdomain II, r_o^I: external radius of subdomain I, r_i^{II}: internal radius of subdomain II; N_b: number of boundary points, N_i: number of internal points, er: maximum relative error of velocity in comparison with analytical results, er_1: maximum difference between single domain and two-domain results, n_i^1: number of iterations for $CM1$, n_i^2: number of iterations for $CM2$ where $CM1$ is the convergence measure for each subdomain and $CM2$ is convergence measure on the artificial boundary.

n	Subdomain I					Subdomain II								
	r_o^I	N_b	N_i	n_i^1	$CM1$	r_i^{II}	N_b	N_i	n_i^1	$CM1$	$CM2$	n_i^2	$er_1\%$	$er\%$
0.7	1.5	72	72	11	10^{-4}	1.5	72	72	6	10^{-4}	$1.13e-2$	6	1.18	1.75
0.5	1.5	72	72	24	10^{-4}	1.5	72	72	10	10^{-4}	$1.24e-2$	9	1.54	2.29
0.4	1.5	72	72	39	10^{-4}	1.5	72	72	14	10^{-4}	$1.37e-2$	11	1.87	2.73

choosing only two subdivisions each of which is a concentric annulus (Fig. 3.2b), it is possible to ensure that magnitudes of the velocity gradients are similar within individual subdomains. The inner annulus has radii of 1 and 1.5 and the outer 1.5 and 2. The data density for each subdomain as well as the results of numerical computations are reported in Table 3.2. As described earlier in section 3.3.1, the mechanism to ensure velocity and traction compatibility at the artificial interfaces is adopted from Davies and Mushtaq (1996) and the degree of compatibility is measured by the convergence measure $CM2$ (Eq. (3.17)).

When $n = 0.3$, 0.25 and 0.2, the variation of the gradient of the velocity field is very strong across the domain. In order to achieve the stated goal that the velocity gradients should be of similar magnitude in each subdomain, the radius of the interface between the two subdomains has to decrease with decreasing n. Of course this process has a limit and as n becomes too small more than two subdomains might be needed. Table 3.3 reports the results corresponding to $n \leq 0.35$.

From Tables 3.1-3.3, it can be seen that for two sub-domains, the accuracy of solutions is not as good as the single-domain results. However, at lower power-

Table 3.3: BEM-MQ-RBFN for circular Couette flow of power law fluids: maximum relative error of velocity field (two subdomains) in comparison with analytical results for different power law indices n. $r_i^I = R_i = 1.0$: internal radius of subdomain I, $r_o^{II} = R_o = 2.0$: external radius of subdomain II, r_o^I: external radius of subdomain I, r_i^{II}: internal radius of subdomain II; N_b: number of boundary points, N_i: number of internal points, er: maximum relative error of velocity in comparison with analytical results, n_i^1: number of iterations for $CM1$, n_i^2: number of iterations for $CM2$ where $CM1$ is the convergence measure for each subdomain and $CM2$ is convergence measure on the artificial boundary.

n	Subdomain I					Subdomain II							
	r_o	N_b	N_i	n_i^1	$CM1$	r_i	N_b	N_i	n_i^1	$CM1$	$CM2$	n_i^2	$er\%$
0.35	1.3	72	72	21	10^{-4}	1.3	84	84	24	10^{-4}	$1.75e-2$	14	3.11
0.30	1.3	72	72	32	10^{-4}	1.3	84	84	31	10^{-4}	$2.11e-2$	17	3.54
0.25	1.2	72	72	42	10^{-4}	1.2	96	96	47	10^{-4}	$2.76e-2$	21	4.15
0.20	1.2	72	72	57	10^{-4}	1.2	96	96	59	10^{-4}	$3.47e-2$	26	5.18

law indices (e.g. $n \leq 0.4$), the results show that the accuracy is nearly as good as those obtained with single domain for $n = 0.4$ and, more importantly, convergence is achieved with two sub-domains for $n < 0.4$ which is not the case with a single domain. This might be explained by the better NN approximation owing to the AVGDD scheme. Furthermore, it is found that the number of iterations associated with each sub-domain decreases and hence a further speed-up is achieved with parallel computation. Apart from overcoming the loss of convergence at lower values of the power-law index, the computation throughput with parallel DD is also improved significantly in comparison with the cases of single domain. For example, with $n = 0.7$, for the 2-sub-domain case, the parallel DD technique makes throughput increase by about 29.5%. In this section, a comparison between the BEM-MLPN, BEM-MQ-RBFN, BEM-TPS-RBFN1 and BEM-TPS-RBFN2 approaches is also carried out for the case of power-law indices $n = 0.7$, $n = 0.4$ (single domain) and $n = 0.2$ (two sub-domains). Table 3.4 shows that at the high power-law index, the MLPN method gives slightly better results in comparison with those by the MQ-RBFN and TPS-RBFN1 methods. However, the convergence rate of the MLPN method is poor in general and as the power-law index decreases the accuracy and convergence

Table 3.4: BEM-MLPN, BEM-MQ-RBFN, BEM-TPS-RBFN1 and BEM-TPS-RBFN2 methods for circular Couette flow of power law fluids: the comparison of accuracy for power-law indices $n = 0.7$ and $n = 0.4$ (single domain) and $n = 0.2$ (two sub-domains). The number of boundary points, the number of internal points, the convergence measure $CM1$ and other parameters are given in Table 3.1 and Table 3.3. er: maximum relative error of the velocity field in comparison with analytical results; n_i: number of iterations; $CM2$ is the convergence measure on the artificial boundary; n_i^1: number of iterations for $CM1$; n_i^2: number of iterations for $CM2$.

Circular	Single domain				Two subdomains				
Couette	$n = 0.7$		$n = 0.4$		$n = 0.2$				
Flow	$er\%$	n_i	$er\%$	n_i	$CM2$	$er\%$	n_i^1	n_i^2	n_i
BEM-MLPN	0.97	16	2.42	61	...	...	...	...	...
BEM-MQ-RBFN	1.16	11	2.57	45	$3.34e-2$	5.18	57	59	26
BEM-TPS-RBFN1	1.22	11	2.71	42	$2.71e-2$	5.97	56	57	26
BEM-TPS-RBFN2	0.91	11	2.27	42	$1.86e-2$	4.83	56	57	26

deteriorate. The present results show that the performance of the MQ-RBFN method is slightly better than that of the TPS-RBFN1 method. However, when the value of m is increased to 2 in the TPS-RBF (3.38), resulting in the TPS-RBFN2 method, the accuracy of the solution and the smoothness on the artificial boundary (in the case of domain decomposition) are improved slightly on the MQ-RBFN method (Table 3.4). Since the accuracy of the approximation can be very dependent on the width of the MQ-RBF, a thorough comparison between the MQ-RBFN and the TPS-RBFN approaches is beyond the scope of the present work.

Carreau-Yasuda and Cross models

A number real fluids were described by these models. For example, the Carreau-Yasuda model parameters for three polymer solutions and three polymer melts are taken from Bird *et al.* (1987a) and presented in Table 3.5. For these models it is found that convergence is achieved with a single domain where 72 boundary points and 144 internal points are used in the case of polystyrene in Aroclor, polyacrylamide in water and glycerin, and 84 boundary points and 168 internal points in the other cases. The tolerance for the $CM1$ is set at $5.e-5$. Table 3.6 shows that the convergence rate depends on both n and λ where it takes 497, 20 and 98 iterations respectively for the 3 cases of polymer solutions and 12, 28, 48 iterations in the case of the 3 polymer melts.

Table 3.5: Carreau-Yasuda model parameters for some polymer solutions and melts, (from Bird *et al.*, 1987a, pp171-175).

Materials	$\eta_o(Pa.s)$	$\eta_\infty(Pa.s)$	$\lambda(s)$	n	a
Polymer solution					
polyisobutylene in Primol 355	9.23e+2	1.50e-1	191	0.358	2
polystyrene in Aroclor	1.01e+2	5.90e-2	0.84	0.364	2
polyacrylamide in water and glycerin	1.06e+1	1.00e-2	8.04	0.364	2
Polymer Melts					
Polystyrene at 453 K	1.48e+4	≈0	1.04	0.938	2
High-density polyethylene 443 K	8.92e+3	≈0	1.58	0.496	2
Phenoxy-A at 485 K	1.24e+4	≈0	7.44	0.728	2

Table 3.6: BEM-MLPN for circular Couette flow of some Carreau-Yasuda fluids: the number of iterations required to reach a convergence measure $CM1$ of $O(10^{-4})$. N_b: number of boundary points, N_i: number of internal points, n_i: number of iterations. Model parameters are given in Table 3.5.

Materials	N_b	N_i	n_i	$CM1$
Polymer solutions				
polyisobutylene in Primol 355	84	168	497	$5.e-5$
polystyrene in Aroclor	72	144	20	$5.e-5$
polyacrylamide in water and glycerin	72	144	98	$5.e-5$
Polymer melts				
Polystyrene at 453 K	72	144	12	$5.e-5$
High-density polyethylene 443 K	72	144	28	$5.e-5$
Phenoxy-A at 485 K	72	144	48	$5.e-5$

Table 3.7: Cross model parameters η_o, η_∞, $K(s)$, n for several materials (from Macosko, 1994, p86).

Materials	$\eta_o(Pa.s)$	$\eta_\infty(Pa.s)$	$K(s)$	n
ABS (200^oC)	45000	-	2.5	0.400
Blood	0.125	0.005	52.5	0.285
Yogurt	10	0.004	0.26	0.100

Table 3.8: BEM-MLPN for circular Couette flow of some Cross fluids: the number of iterations required to reach a convergence measure $CM1$ of $O(10^{-4})$. N_b: number of boundary points, N_i: number of internal points, n_i: number of iterations. Model parameters are given in Table 3.7.

Materials	N_b	N_i	n_i	$CM1$
ABS (200^oC)	72	144	48	$5.e-5$
Blood	72	144	522	$5.e-5$
Yogurt	72	144	23	$5.e-5$

The Cross model can be used to describe several materials whose parameters are given in Table 3.7 which is taken from Macosko (1994). Table 3.8 reports the results of the analysis of these materials.

The radial profile of the velocity field corresponding to both models is similar to the power-law profile in terms of their general shape (Fig. 3.3) and therefore is not specifically shown here.

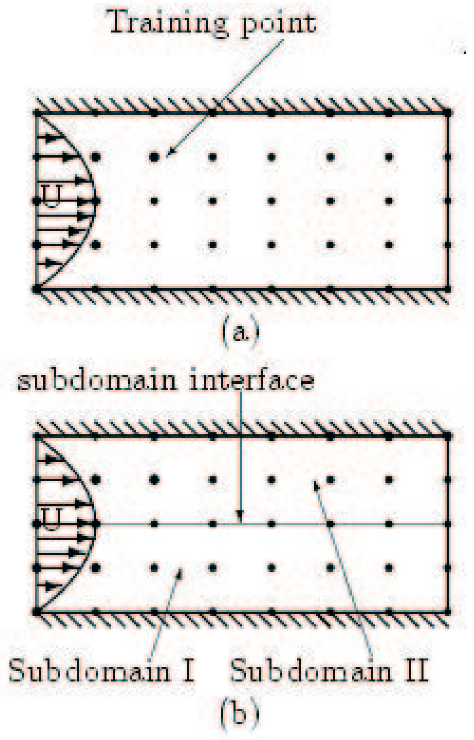


Figure 3.4: Planar Poiseuille flow problem: (a) single domain; (b) two subdomains. The velocity profile is assumed to be parabolic and the walls to be non-slip. Training point distribution is only schematic.

3.4.2 Planar Poiseuille problem

The planar Poiseuille flow problem is described in Fig 3.4. Let $2a$ be the gap between the two parallel plates and a be the length of the domain under consideration. A typical shear rate is U/a where U is the centreline fluid speed, $\eta_* \frac{U}{a}$ is a typical stress mentioned in §3.3.2. The inlet and outlet boundary conditions are assumed to be unidirectional velocity with a parabolic profile. The solution shows that the parabolic profile is reproduced throughout the domain as expected. The present method is further verified by solving the problem with the constant pressure drop boundary condition instead of the velocity one as shown in Fig. 3.5 where the pressure drop is $\Delta p = 1$. With all other parameters the same, the velocity profile is obtained as shown in Fig. 3.6 for the power-law model with a single domain. It can be seen in Fig. 3.6 that the present numer-

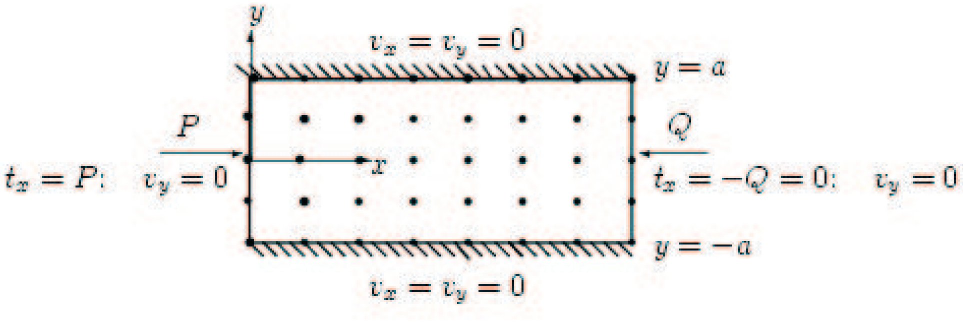


Figure 3.5: Planar Poiseuille flow problem: The pressure drop $\Delta p = (P - Q) > 0$ and the no-slip wall boundary conditions. Training point distribution is only schematic.

ical results are accurate in comparison with the exact solution which is given by

$$v_x = \frac{n}{n+1}\left(\frac{\Delta p}{m}\right)^{1/n}\left(a^{\frac{n+1}{n}} - |y|^{\frac{n+1}{n}}\right), \tag{3.43}$$

where the coordinates (x, y) are defined in Fig. 3.5.

Power-law model

Similar to the case of Couette flow of power-law fluid, it is found that the convergence of the procedure is difficult to achieve at lower power-law indices with a single domain. However, with DD convergence can be easily achieved.

Results for the single domain case (for $n = 0.7, 0.6, 0.5$)

The number of boundary points and internal points are 52 and 135 respectively distributed on an 11×17 grid for $n = 0.7, n = 0.6$. For $n = 0.5$, the corresponding points are 60 and 171 on an 11×21 grid. Table 3.9 illustrates the influence of the power-law index n on the convergence rate of the present BEM-MLPN method.

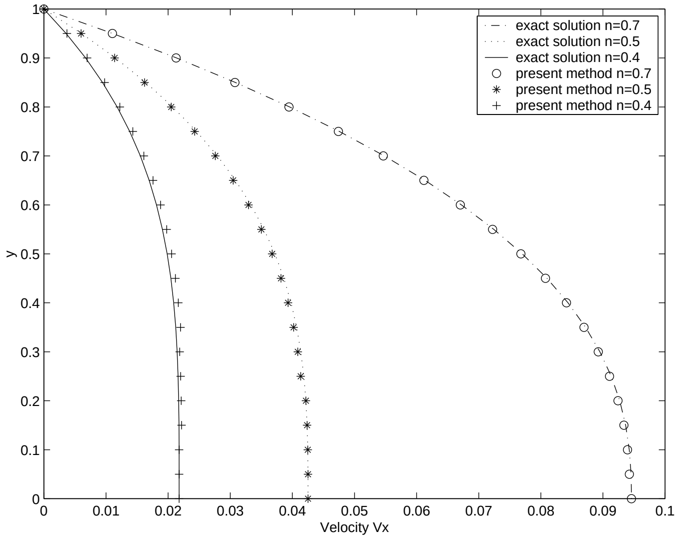


Figure 3.6: Planar Poiseuille flow of power-law fluid with pressure drop boundary conditions: comparison of the present BEM-RBFN results with the exact (analytical) solution where the velocity profiles correspond to several values of the power-law index n.

Results for the 2-subdomain case (for $n \leq 0.4$)

The analysis domain is divided into two subdomains symmetrical about the centreline. The number of data points per each subdomain is shown in Table 3.10 where the convergence characteristics are associated with the BEM-MQ-RBFN approach. The corresponding results associated with the BEM-MLPN method are not shown. However, it is noted that the training of the MLPNs are less deterministic due to the nature of the BP algorithm. Specifically, al-

Table 3.9: BEM-MLPN for planar Poiseuille flow of power law fluids (single domain): the number of iterations required to reach a convergence measure $CM1$ of $O(10^{-4})$ for different power law indices. N_b: number of boundary points, N_i: number of internal points, n_i: number of iterations.

n	N_b	N_i	n_i	$CM1$
0.7	52	135	37	$5.e-5$
0.6	52	135	41	$5.e-5$
0.5	60	171	98	$5.e-5$

Table 3.10: BEM-MQ-RBFN for planar Poiseuille flow of power law fluids (two subdomains): the number of iterations required to reach a convergence measure $CM1$ of $O(10^{-4})$ for different power law indices. N_b: number of boundary points, N_i: number of internal points, n_i^1: number of iterations for $CM1$, n_i^2: number of iterations for $CM2$ (CM1 is the convergence measure for each subdomain and $CM2$ the convergence measure on the artificial boundary). The width of the channel is $l_1 + l_2$.

n	Subdomain I					Subdomain II						
	l_1	N_b	N_i	n_i^1	$CM1$	l_2	N_b	N_i	n_i^1	$CM1$	$CM2$	n_i^2
0.4	1.0	44	99	38	$5.e-5$	1.0	44	99	38	$5.e-5$	10^{-3}	19
0.3	1.0	48	117	81	$5.e-5$	1.0	48	117	81	$5.e-5$	10^{-3}	25
0.25	1.0	52	135	97	$5.e-5$	1.0	52	135	97	$5.e-5$	10^{-3}	27

Table 3.11: BEM-MQ-RBFN for planar Poiseuille flow of some Carreau-Yasuda fluids (single domain): the number of iterations required to reach a convergence measure $CM1$ of $O(10^{-4})$. N_b: number of boundary points, N_i: number of internal points, n_i: number of iterations. Model parameters are given in Table 3.5.

Materials	N_b	N_i	n_i	$CM1$
Polymer melts				
Polystyrene at 453 K	60	171	17	$5.e-5$
High-density polyethylene 443 K	60	171	31	$5.e-5$
Phenoxy-A at 485 K	60	171	55	$5.e-5$

though the two subdomains are identical owing to symmetry, the training of the corresponding MLPNs requires very different number of iterations for the same convergence criterion in contrast to the case of RBFNs where both subdomains require the same numbers of iterations.

Carreau-Yasuda and Cross models

As in the case of the circular Couette geometry, the planar Poiseuille flow is successfully simulated with a single domain for both the Carreau-Yasuda and the Cross models. The number of data points is the same in both cases and

Table 3.12: BEM-MQ-RBFN for planar Poiseuille flow of some Cross fluids (single domain): the number of iterations required to reach a convergence measure $CM1$ of $O(10^{-4})$. N_b: number of boundary points, N_i: number of internal points, n_i: number of iterations. Model parameters are given in Table 3.7.

Materials	N_b	N_i	n_i	$CM1$
ABS (200^oC)	60	171	54	$5.e-5$
Blood	60	171	492	$5.e-5$
Yogurt	60	171	21	$5.e-5$

distributed on a 21 × 11 grid. The results for three polymer melts described by the Carreau-Yasuda model (Table 3.5) are shown in Table 3.11. On the other hand, the Cross model is used to describe three common materials (Table 3.7) which are simulated with results shown in Table 3.12.

3.4.3 'Mesh convergence'

The results reported in the preceeding sections correspond to the highest data densities. The confidence in the results is established with a series of increasing densities and 'mesh convergence' is then measured by the following criterion

$$CR = \sqrt{\frac{\sum_{ITP}\sum_{i=1}^{2}(u_i^n - u_i^{n-1})^2}{\sum_{ITP}\sum_{i=1}^{2}(u_i^n)^2}}, \tag{3.44}$$

where ITP is the number of internal test points, u_i^{n-1} is the ith component of the velocity at an internal test point belonging to the coarser grid and u_i^n is the corresponding quantity associated with the finer grid. For example, the circular Couette flow of polyisobutylene in Primol 355 (Table 3.5) described by the Carreau-Yasuda model is analyzed with the following seven different grid densities (number of boundary points, number of internal point)= (36,72), (44,88), (52,104), (60,120), (68,136), (72,144) and (84,168) with a set of 120 internal test points and Table 3.13 reports the trend of CR with increasing grid density. The process is deemed to have achieved 'mesh convergence' when CR is $O(10^{-3})$.

Table 3.13: BEM-MLPN simulation of the circular Couette flow of polyisobutylene in Primol 355 (described by the Carreau-Yasuda model): Trend of the 'mesh convergence' measure, CR defined by (3.44) with increasing number of training points. N_b: number of boundary points, N_i: number of internal points, TP: number of training points. The number of internal test points is 120.

N_b	N_i	TP	CR
36	72	108	1.0000
44	88	132	0.3113
52	104	156	0.1076
60	120	180	0.0350
68	136	204	0.0187
72	144	216	0.0105
84	168	252	0.0094

3.5 Concluding remarks

A boundary-element-only method is formulated and implemented for the flow analysis of GNFs. The BEM-NN approach results from the use of NNs for the global approximation of field variables and a particular solution technique to estimate the volume integral. Volume discretisation is avoided and new linear boundary element techniques are developed where a general treatment of the corner problem and exact calculations of singular integrals are achieved. The present approach is a first step towards element-free approaches, developed in subsequent chapters, for the simulation of non-Newtonian fluid flows. The method is able to achieve convergence for the difficult situation of the flow of power-law fluids at small power-law indices (as low as 0.2). Although the Couette and planar Poiseuille flow are geometrically simple, the material is highly non-linear and convergence and accuracy achieved here were not achieved elsewhere previously. Thus, in Table 3.1 (Couette flow), relative errors are given as 0.97%, 1.13%, 1.65%, and 2.42% corresponding to power law indices 0.7, 0.6, 0.5, and 0.4, respectively, which are relatively small in comparison with others (e.g. Florez et al., 2000). Similarly, the maximum error in the case of the Planar Poiseuille flow (Figure 3.6) is 2.87% corresponding to the worst case of n=0.4.

Furthermore, the present method achieved convergence for indices as low as 0.2 where no other methods could achieve before. The present combination of BEM and MLPN has demonstrably achieved better results than those obtained by BEM alone and so the error trend was due to increasing non-linearity (decreasing power law index), not by BEM or MLPN. The accuracy depends on the number of collocation points N as already presented in section 3.4.3, Table 3.13, where convergence quality was shown to increase with increasing N. In the next chapters, the coupling of the RBFN-based element-free method and the Brownian dynamics simulation will be described for dilute polymer solutions. Specifically, chapters 4 and 5 are devoted to the Lagrangian approach and chapter 6 and 7 report the development of the Eulerian method.

Chapter 4

Calculation of Viscoelastic Fluid Flow Using Neural Networks and Stochastic Simulation (CVFNNSS)

This chapter reports a new technique, from the Lagrangian point of view, for numerical calculation of viscoelastic flows based on the combination of a RBFN-based mesh-free method and Brownian dynamics simulation or stochastic simulation technique (Tran-Canh and Tran-Cong, 2002b). This method uses an 'universal approximator' based on neural network methodology in combination with the kinetic theory of polymeric liquid in which the stress is computed from the molecular configuration rather than from closed form constitutive equations. Thus the present method obviates not only the need for a rheological constitutive equation to describe the fluid (as in the original CONNFESSIT idea) but also any kind of finite element-type discretisation of the domain and its boundary for numerical solution of the governing PDEs.

4.1 Introduction

The computation of viscoelastic fluid flow has undergone strong development for the last three decades or so. Most common methods of numerical computation and analysis are macroscopic in nature where the system of mass and momentum conservation equations are supplemented by an appropriate closed form constitutive equation. The disadvantage appears for those models that cannot be cast into a closed form. A technique of hybrid simulation using Brownian dynamics, namely the CONNFFESSIT proposed by Laso and Öttinger (1993), has been introduced to bypass the need of a closed form constitutive equation. It is a combination of the traditional element method and the SST. The main idea of the CONNFFESSIT approach is that the polymer contribution to the stress is calculated from the configuration of a large ensemble of microscopic entities, which acts as a stress calculator instead of a closed form constitutive equation (Öttinger, 1996; Laso and Öttinger, 1993; Feigl *et al.*, 1995; Laso *et al.*, 1997, 1999). This approach is an attempt to emulate the situation in real liquids, where the full information about the stress is contained in the configuration of molecules which results from the deformation history. However, the FEM in CONNFFESSIT requires the discretisation of the domain under consideration into a number of finite elements (FE) which are defined by certain fixed topology in terms of a number of nodes. Breaking the original domain of analysis into a set of finite elements is not easy, specially for problems with moving boundaries, complex boundary or free surface. In addition to the popular methods for the numerical solution of PDEs such as FEM, BEM and FVM, more recent NN-based methods such as RBFNs (Kansa, 1990b; Zerroukat *et al.*, 1998; Mai-Duy and Tran-Cong, 2001), MPLNs (He *et al*, 2000), approximate identity networks (AINs) (Conti and Turchetti, 1994) prove to be promising. Such a NN-based method is developed in conjunction with the Brownian dynamics simulation from the Lagrangian point of views in the present chapter. Thus in contrast to FE-type approximations, the presently proposed Computation of Viscoelastic Flow by NN and SS method (CVFNNSS) is based on a direct combination of the stochastic simulation of molecular model of polymers with RBFN-based element-free techniques. In the present CVFNNSS method, the polymer stress is computed by a Brownian simulation technique as a component of the macromolecular approach (Fixman, 1978a,b; Bird *et al.*, 1987; Öttinger, 1996). The polymer-contributed stress is then used as position-dependent known terms in solving the continuity and momentum equations in a macroscopic approach. The present method does not require any fixed connectivity to satisfy a predetermined topology, i.e. a mesh in which the elements are constrained by some geometrical regularity conditions (e.g. a positive volume). The present discrete model is completely represented by a set of unstructured discrete collocation nodes in the analysis domain and on its boundary in both microscopic and macroscopic part of the CVFNNSS procedure and therefore is referred to as a mesh-free numerical technique according to commonly cited concepts (Onate *et al.*, 1996; Belytschko *et al.*, 1996) and mentioned in §1.4.1. This chapter is organized as follows. In §4.2, §4.3, §4.4, the basic ideas of Lagrangian CVFNNSS are presented in which the governing equations and the stochastic simulation technique for computing the stress are described. In §4.5, the NN-based numerical method for approximation of a function and its derivatives is reviewed briefly and the RBFN method for solving the conservation equations is described. A treatment of the time dependent ordinary differential equation is a major topic of this chapter. §4.6 presents the algorithm of the CVFNNSS procedure, highlighting the macroscopic-microscopic interfaces of the method. The numerical examples are then discussed in §4.7 with a brief conclusion in §4.8.

4.2 Governing Equations

Consider the transient isothermal flow of an incompressible fluid with density ρ, the system of momentum and mass conservation equations is given by

$$\rho\frac{\partial \mathbf{u}}{\partial t} + \rho(\mathbf{u}\cdot\nabla)\mathbf{u} = -\nabla p + \nabla\cdot\boldsymbol{\tau}, \tag{4.1}$$

$$\nabla\cdot\mathbf{u} = 0, \tag{4.2}$$

where p is the pressure arisen from the incompressibility constraint; $\mathbf{u}$ denotes velocity field; $\boldsymbol{\tau}$ is the extra stress. As mentioned in the previous chapters, the extra stress is then further decomposed as

$$\boldsymbol{\tau} = 2\eta_N\mathbf{D} + \boldsymbol{\tau}^p, \tag{4.3}$$

where $2\eta_N\mathbf{D}$ is the Newtonian solvent contribution; η_N is the solvent viscosity; $\mathbf{D}$ is the rate of strain tensor; $\boldsymbol{\tau}^p$ is the polymer-contributed stress. Using Eq (4.3), Eq (4.1) can be rewritten as follows

$$\rho\frac{\partial \mathbf{u}}{\partial t} + \rho(\mathbf{u}\cdot\boldsymbol{\nabla})\mathbf{u} = -\boldsymbol{\nabla} p + \boldsymbol{\nabla}\cdot(2\eta_s\mathbf{D} + \boldsymbol{\tau}^p). \tag{4.4}$$

In the traditional macroscopic approach the sytem is usually closed by the specification of a closed form constitutive equation for the polymer-contributed stress $\boldsymbol{\tau}^p$. In contrast, $\boldsymbol{\tau}^p$ is here calculated numerically via a microscopic technique. The overall macro-microscopic procedure is described in the next section.

4.3 The micro-macroscopic approach

The micro-macroscopic approach employed in this thesis uses Brownian dynamics simulation (or stochastic simulation technique) (see §1.3, chapter 1) and the general procedure for this approach is as follows. At each time step the polymer-contributed stress is assumed known from the previous iteration and the system of mass and momentum conservation equations is solved by a macroscopic numerical method. The velocity field thus obtained is then used in a stochastic simulation technique to calculate the polymer-contributed stress. The iteration is continued until convergence is achieved before advancing to the next time level.

The microscopic method employs the Brownian dynamics simulation to determine the stress via kinetic modelling which is described in the next section, followed by the description of a mesh-free collocation method for the solution of the continuity and momentum equations. All function approximations are based on RBFNs and the overall procedure is free of finite-element-type discretisation and thus referred to as mesh-free.

In the next section, the stochastic simulation technique is presented for the computation of the polymer-contributed stress $\boldsymbol{\tau}^p$.

4.4 Stochastic simulation technique in polymeric kinetic theory

As mentioned in chapter 1, in polymer kinetic theory, the determination of polymer stresses is carried out through two steps. The first step is to derive the diffusion equation or Fokker-Planck equation for the configurational distribution function $\psi(\mathbf{Q}, t)$ which is the probability density of the polymer configuration $\mathbf{Q}$ occurring at time t. The second step is to develop an expression for the stress tensor corresponding to the polymer configuration of which the distribution is expressed by $\psi(\mathbf{Q}, t)$. The stochastic simulation is based on the relationship between the diffusion equation and the stochastic differential equation (SDE).

In the present work, two nonlinear dumbbell models, the FENE and FENE-P models, are used in the development of a mesh-free stochastic simulation technique. The FENE-P has a corresponding approximate closed form constitutive equation (Bird *et al.*, 1987b) while, the FENE model has not. These two dumb-

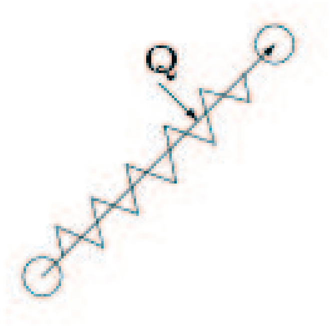


Figure 4.1: Elastic dumbbell model: the connector vector **Q** describes the configuration of the model.

bell models are shown in Fig. 4.1 where the polymer configuration is described by the connector vector $\mathbf{Q}(t)$. The dynamics of polymeric liquids can be represented by the diffusion equation for $\psi(\mathbf{Q}, t)$ which, in the absence of external forces, is rewritten from (1.14) as follows

$$\frac{\partial}{\partial t}\psi(\mathbf{Q},t) = -\frac{\partial}{\partial \mathbf{Q}} \cdot [\mathbf{A}(\mathbf{Q},t)\psi(\mathbf{Q},t)] + \frac{1}{2}\frac{\partial}{\partial \mathbf{Q}} \cdot \left[\frac{\partial}{\partial \mathbf{Q}} \cdot \{\mathbf{L}(\mathbf{Q},t)\psi(\mathbf{Q},t)\}\right], \quad (4.5)$$

where $\mathbf{A}(\mathbf{Q}, t)$ is a 3-component column vector; $\mathbf{L}(\mathbf{Q}, t)$ is a positive semidefinite 3×3 matrix. Instead of solving Eq (4.5) directly, the polymer configuration $\mathbf{Q}(t)$ is determined by using an equivalent SDE which is given by

$$d\mathbf{Q}(t) = \mathbf{A}(\mathbf{Q}(t), t)dt + \mathbf{B}(\mathbf{Q}(t), t) \cdot d\mathbf{W}(t), \quad (4.6)$$

where $\mathbf{W}(t)$ is a 3-component column vector which is a Wiener process with mean $\langle W_i(t)\rangle = 0$ and covariance $\langle W_i(t)W_j(t')\rangle = \delta_{ij}\min(t, t')$; $\mathbf{B}(\mathbf{Q}, t)$ is a 3×3 matrix and $\mathbf{L}(\mathbf{Q}, t) = \mathbf{B}(\mathbf{Q}, t)\mathbf{B}^T(\mathbf{Q}, t)$. Stochastic theory shows that in general, the tensor **B** is existent but not unique (Öttinger, 1996). There are several schemes to evaluate the tensor **B**, among them is the Cholesky decomposition which is generally employed. In the present work, **B** is specifically given in closed form for each of the FENE and FENE-P models as shown in later sections. The theory also shows that although the trajectories $\mathbf{Q}(t)$ obtained from the SDE are different for various choices of the tensor **B**, all transition probabilities $\psi(\mathbf{Q}, t)$ and then averages of $\mathbf{Q}(t)$ are identical (more details can be found in Öttinger, 1996, for example).

In simulating the flow of polymeric fluids, for a given kinetic model of polymer molecules, the solution of Eq (4.6) at each time t_i (the time discretization) is obtained by the simulation of the configuration of the model, which starts from a given probability distribution function of the configuration at t_o. The drift and diffusion terms **A** and **B** respectively of Eq (4.6) depend on the given kinetic models. The numerical integration of the SDE (4.6) can be carried out using different schemes (see §2.3.2, chapter 2). In this chapter, the explicit Euler integration scheme is employed. The SDEs and their numerical solution for the FENE and FENE-P models are presented in the next section.

4.4.1 FENE dumbbell model

The configuration of a dumbbell is completely described by the length and orientation of the vector **Q** connecting the two beads (Fig. 4.1). In a Hookean dumbbell model the linear spring force is realistic only for small deformation from the static equilibrium configuration and the extent of the dumbbell's stretch is not limited. This unphysical behaviour is removed in the FENE model which plays an important role in the description of non-linear rheological phenomena. Neglecting the external forces, the diffusion equation (4.5) corresponding to the FENE model can be expressed as (Öttinger, 1996; Bird *et al.*, 1987b)

$$\frac{\partial}{\partial t}\psi(\mathbf{Q},t) = \frac{2}{\zeta}\frac{\partial}{\partial \mathbf{Q}} \cdot \mathbf{F}\psi - \frac{\partial}{\partial \mathbf{Q}} \cdot [\boldsymbol{\kappa} \cdot \mathbf{Q}]\psi + \frac{2k_BT}{\zeta}\frac{\partial}{\partial \mathbf{Q}} \cdot \frac{\partial}{\partial \mathbf{Q}}\psi, \qquad (4.7)$$

where $\boldsymbol{\kappa}$ is the velocity gradient which can be a function of time, but not position (i.e. locally homogenous flows at the dumbbell (Bird *et al.* 1987b, §13.2)). The velocity gradient tensor is calculated analytically from the velocity field which is approximated by TPS-RBFNs (see §4.5 and step (d) of §4.6 for more details). In the present work, the local homogeneity of the flow around each dumbbell can naturally be assumed in an arbitrarily small volume around the dumbbell. The spatially constant velocity gradient takes the value computed at the dumbbell centre of mass position. The FENE spring force **F** is given by (Bird *et al.*, 1987b; Herrchen and Öttinger, 1997)

$$\mathbf{F} = \frac{H}{1 - \left(\frac{\mathbf{Q}}{Q_o}\right)^2}\mathbf{Q}, \tag{4.8}$$

where Q^o is the maximum possible spring length. The SDE corresponding to the FENE model is now given by

$$d\mathbf{Q}(t) = \left(\boldsymbol{\kappa}(t) \cdot \mathbf{Q}(t) - \frac{2H}{\zeta}\frac{\mathbf{Q}(t)}{1 - \left(\frac{\mathbf{Q}(t)}{Q_o}\right)^2}\right) dt + \sqrt{\frac{4k_BT}{\zeta}}d\mathbf{W}(t), \tag{4.9}$$

where $\mathbf{W}(t)$ accounts for the random displacement of the beads due to thermal motion; T is the absolute temperature and k_B is Boltzmann constant.

Let $\mathbf{Q}_i$ be an approximation of $\mathbf{Q}(t)$ at time t_i, $\Delta t_i = t_{i+1} - t_i$. The solution from the explicit Euler integration scheme is written as follows

$$\mathbf{Q}_{(i+1)} = \mathbf{Q}_i + \left(\boldsymbol{\kappa}_i \cdot \mathbf{Q}_i - \frac{2H}{\zeta}\frac{\mathbf{Q}_i}{1 - \left(\frac{\mathbf{Q}_i}{Q_o}\right)^2}\right) \Delta t_i + \sqrt{\frac{4k_BT}{\zeta}}\Delta\mathbf{W}_i. \tag{4.10}$$

The dimensionless forms of equations (4.9) and (4.10) are written as follows (by dividing by $\sqrt{\frac{k_BT}{H}}$)

$$d\mathbf{Q}'(t) = \left(\boldsymbol{\kappa}(t) \cdot \mathbf{Q}'(t) - \frac{1}{2\lambda_H}\frac{\mathbf{Q}'(t)}{1 - \frac{\mathbf{Q}'^2}{b}}\right) dt + \sqrt{\frac{1}{\lambda_H}}d\mathbf{W}(t), \tag{4.11}$$

$$\mathbf{Q}'_{(i+1)} = \mathbf{Q}'_i + \left(\boldsymbol{\kappa}_i \cdot \mathbf{Q}'_i - \frac{1}{2\lambda_H}\frac{\mathbf{Q}'_i}{1 - \frac{\mathbf{Q}'^2_i}{b}}\right) \Delta t_i + \sqrt{\frac{\Delta t_i}{\lambda_H}}\mathbf{W}_i, \tag{4.12}$$

where $\mathbf{Q}'_i = \mathbf{Q}_i[H/(k_BT)]^{1/2}$ is the dimensionless connector vector at t_i; $\lambda_H = \zeta/(4H)$ is the relaxation time of dumbbells; $b = \frac{HQ_o^2}{k_BT}$ is the square of the maximum extension of the dimensionless connector vector $\mathbf{Q}'$. Thus in the FENE dumbell model, the connector vector cannot be stretched beyond a maximum value of $\sqrt{b}$ in the dimensionless sense (Laso and Öttinger, 1993). The components of the random vector $\mathbf{W}_i$ are independent Gaussian variables with mean zero and variance Δt (Here the time step size is fixed $\Delta t_i = \Delta t$). This model has no corresponding closed form constitutive equation for the polymeric stress tensor and therefore it cannot be solved with the traditional macroscopic approaches.

Based on the polymer configuration Eq (4.12), the dimensionless form of the polymer contributed stress tensor at the time t_i can be determined as follows (Öttinger, 1996; Bird *et al.*, 1987b; Herrchen and Öttinger, 1997)

$$\boldsymbol{\tau}_i^{'p} = -n_d k_B T \left(\left\langle \frac{\mathbf{Q'}_i \mathbf{Q'}_i}{1 - \frac{\mathbf{Q}_i'^2}{b}} \right\rangle - \mathbf{I} \right), \tag{4.13}$$

where n_d is number of dumbbells per unit volume of the solution.

4.4.2 FENE-Peterlin (FENE-P) dumbbell model

The FENE-P model is based on the FENE dumbbell model in which the term $\frac{\mathbf{Q^2}}{Q_o^2}$ in the denominator of the spring force shown in equation (4.8) is replaced by its everage $\langle\frac{\mathbf{Q^2}}{Q_o^2}\rangle$. The spring force $\mathbf{F}$ is rewritten as follows (Bird *et al.*, 1987b; Herrchen and Öttinger, 1997; Keunings, 1996)

$$\mathbf{F} = \frac{H}{1 - \langle\frac{\mathbf{Q}^2}{Q_o^2}\rangle}\mathbf{Q}. \tag{4.14}$$

In this case, the dimensionless form of the polymer configuration and the contribution to stress are given as follows, respectively,

$$\mathbf{Q}'_{(i+1)} = \mathbf{Q}'_i + \left(\boldsymbol{\kappa}_i \cdot \mathbf{Q}'_i - \frac{1}{2\lambda_H} \frac{\mathbf{Q}'_i}{1 - \frac{\langle \mathbf{Q}_i'^2 \rangle}{b}} \right) \Delta t_i + \sqrt{\frac{\Delta t_i}{\lambda_H}} \mathbf{W}_i, \tag{4.15}$$

$$\boldsymbol{\tau}_i^{'p} = -n_d k_B T \left(\frac{\langle \mathbf{Q}'_i \mathbf{Q}'_i \rangle}{1 - \frac{\langle \mathbf{Q}_i'^2 \rangle}{b}} - \mathbf{I} \right). \tag{4.16}$$

The average which appears in the denominator of equations (4.15) and (4.16) is calculated over the number of dumbbells in a small local domain where the dumbbells are located. From equation (4.15), it can be seen that the length of connector vectors $\mathbf{Q}'$ could become greater than the maximum allowable limit $\sqrt{b}$ during the simulation (Laso and Öttinger, 1993; Keunings, 1996).

In the present work, this unphysical situation is corrected by contracting the unphysical value as follows $\|\mathbf{Q}'\| = \|\mathbf{Q}'\| - \mod(\|\mathbf{Q}'\|, \sqrt{b})$. The 'reflecting' method of Laso and Öttinger (1993) is a special case in which the length of $\mathbf{Q}'$ satisfies: $\sqrt{b} < \|\mathbf{Q}'\| < 2\sqrt{b}$.

At time step $(i+1)$, the computed stress tensor $\boldsymbol{\tau}^{p}_{i+1}$ is then employed to get the solution of the velocity field $\mathbf{u}$ from the governing PDEs (4.1) and (4.2) which are solved by an element-free RBFN-based numerical method presented in the next section.

4.5 RBFNs for solving the continuity and momentum equations

Recently, the application of RBFNs in numerical solution of PDEs have brought interesting results (Kansa, 1990; Zerroukat *et al.*, 1998; Mai-Duy and Tran-Cong, 2001). Comparing many available interpolation methods for scattered data, Franke (1982) ranked Multiquadric RBF(MQ-RBF) of Hardy (1971) and Thin Plate Splines RBF (TPS-RBF) of Duchon (1976) as superior in accuracy and both of these RBFs are employed in this chapter.

4.5.1 Review of Radial basis function network interpolation

As presented in the chapter 3, the present work uses the neural-like network with one hidden layer of RBFs where the function $f(\mathbf{x})$ is decomposed into m fixed RBFs as

$$f(\mathbf{x}) = \sum_{j=1}^{m} w^j h^j(\mathbf{x}), \tag{4.17}$$

where w^j is the synaptic weight and h^j is the chosen radial basis function corresponding to the j^{th} neuron). Usually $m \leq n$ (Haykin, 1999) where n is the number of input data points $(\mathbf{x}_i, \hat{y}_i)$; $\mathbf{x}_i$ is the coordinate of the i^{th} collocation point ($\mathbf{x}_i$ is a scalar in the case of 1-dimensional space) and $\hat{y}_i$ is the desired value of function f at the collocation point $\mathbf{x}_i$. The partial derivatives of $f(\mathbf{x})$ can be calculated analytically as follows

$$\frac{\partial^k f(\mathbf{x})}{\partial x_i ... \partial x_n} = \sum_{j=1}^{m} w^j \frac{\partial^k h^j}{\partial x_i ... \partial x_n}, \tag{4.18}$$

where the RBF's h^j employed here are either MQ-RBF or TPS-RBF. Together with the first order derivatives (3.35) and (3.38) described in chapter 3, the corresponding second order derivatives of MQ-RBF and TPS-RBF are given respectively by

$$\begin{aligned}
\frac{\partial^2 h^j}{\partial x_i \partial x_l} &= \frac{\chi(r) - (x_i - c_i^j)(x_l - c_l^j)}{\sqrt{(r^2 + (b^j)^2)^3}}, \\
\chi(r) &= r^2 + (b^j)^2 \qquad \forall i = l, \\
\chi(r) &= 0 \quad \forall i \neq l
\end{aligned} \tag{4.19}$$

$$\begin{aligned}
\frac{\partial^2 h^j}{\partial x_i \partial x_l} &= 2r^{2(s-2)}(x_i - c_i^j)(x_l - c_l^j)[2s(s-1)\log(r) + (2s-1)] + \chi(r), \\
\chi(r) &= r^{2(s-1)}(2s\log(r) + 1) \qquad \forall i = l, \\
\chi(r) &= 0 \quad \forall i \neq l
\end{aligned} \tag{4.20}$$

where $\mathbf{r} = (\mathbf{x} - \mathbf{c}^j)$ and $r = \|(\mathbf{x} - \mathbf{c}^j)\|$ is the Euclidean norm of $\mathbf{r}$; $\{\mathbf{c}^j\}$ is a set of centers that can be chosen from among the data points; $b^j > 0$ is the width of the j^{th} RBF.

Since TPS-RBF is C^{2s-1}-continuous, the power index s must be appropriately chosen for a given partial differential operator. In this chapter, the existence of the second order derivatives of unknowns, which need to be approximated in the governing PDE (4.4), requires $m \geq 2$ to satisfy the continuity condition.

The training of the linear model (in w^j) Eq (4.17), given a training set of n collocation points $\{(\mathbf{x}_i, \hat{y}_i)\}_{i=1}^{n}$ can be found in §3.3.6, chapter 3. When the training is completed, the RBFN-based approximation of the function $f(\mathbf{x})$ is given by Eq (4.17) and its derivatives can be calculated analytically in terms of the basis functions according to Eqs (3.35) and (4.18) in the case of MQ-RBFNs or Eqs (3.38) and (4.20) in the case of TPS-RBFNs.

4.5.2 Time integration of the momentum conservation equation by a RBFN-based least-squares method

This section describes the RBFN-based numerical method employed to solve the continuity and the momentum equations in which the velocity field $\mathbf{u}$ is considered as an unknown and the polymer contributed stress $\boldsymbol{\tau}^p$ is already computed by the SST as described in §4.3 and §4.4. Specifically, the start-up planar Couette flow is used to explain the method.

• Governing equation, boundary conditions and initial conditions

For the start-up planar Couette flow problem, a Cartesian coordinate system is chosen as shown in Fig. 4.2. For $t < 0$, the fluid is at rest. At $t = 0$, the lower plate starts to move with a constant velocity V. No-slip condition is assumed at the walls, following Mochimaru (1983). A velocity field that satisfies the equation of continuity (4.2) is given by

$$u = u_x = u_x(y,t), \qquad u_y = 0, \qquad u_z = 0. \tag{4.21}$$

In the present problem, it is not neccessary to calculate the pressure field (Laso and Öttinger, 1993; Mochimaru, 1983) and the momentum equation (4.4) is rewritten as follows

$$\rho\frac{\partial u}{\partial t} = \eta_s\frac{\partial^2 u}{\partial y^2} - \frac{\partial \tau^p_{yx}}{\partial y} \qquad y \in \Omega, \tag{4.22}$$

where y and t are the space and time coordinates; ρ is the density of the fluid; η_s is the solvent viscosity; $\tau^p_{yx}(y,t)$ is the polymer-contributed stress. In the

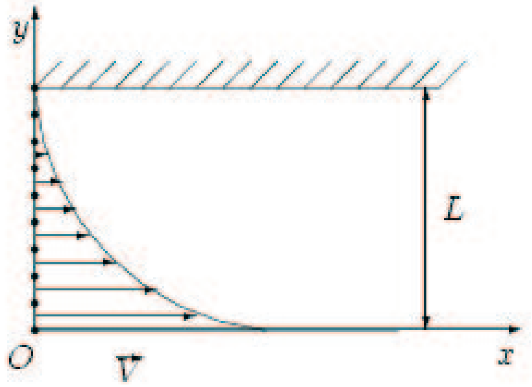


Figure 4.2: The start-up planar Couette flow problem for $t \geq 0$: the bottom plate moves with a constant velocity V, the top plate is fixed; no-slip boundary conditions apply at the fluid-solid interfaces. The collocation point distribution is only schematic.

more general situation the pressure would have to be calculated.

Equation (4.22) is subjected to Dirichlet boundary conditions as follows

$$\begin{aligned} u(0,t) &= V \qquad \forall t > 0, \\ u(L,t) &= 0 \qquad \forall t > 0, \end{aligned} \tag{4.23}$$

and the initial conditions

$$u(0,0) = V, \quad u(y,0) = 0 \qquad \forall y \neq 0. \tag{4.24}$$

The shear stress τ^p_{yx}, at a time step is considered as a known function of y and calculated in the previous step by Brownian simulation technique already described. Its derivative $\frac{\partial \tau^p_{yx}}{\partial y}$ is approximated by a RBFN-based method as presented in §4.5.1.

The PDE (4.22), subject to conditions (4.23) and Eq (4.24), can be solved by a RBFN-based numerical method which is described in the next subsection.

- **Solution of differential equations by a RBFN-based least squares method**

Using the standard implicit approximation and rearranging the terms, equation (4.22) can be written as follows (Kansa, 1990b; Zerroukat *et al.*, 1998; Constantinides and Mostoufi, 1999)

$$u^{n+1} + \alpha \frac{\partial^2 u^{n+1}}{\partial y^2} = u^n + \beta \frac{\partial^2 u^n}{\partial y^2} + \Delta t K^{n+1/2}, \tag{4.25}$$

where Δt is a uniform time step size; let $t_n = t_{n-1} + \Delta t$, $u^n = u(y, t_n)$; $\alpha = -\theta \Delta t \frac{\eta_s}{\rho}$; $\beta = (1-\theta)\Delta t \frac{\eta_s}{\rho}$ with $0 \le \theta \le 1$; $K = \frac{1}{\rho}.\frac{\partial \tau^p_{yx}}{\partial y}$; and (Constantinides and Mostoufi, 1999)

$$K^{n+1/2} = \frac{1}{2}(K^{n+1} + K^n), \tag{4.26}$$

where $K^n = K(t_n)$ and K^{n+1} is approximated by the backward difference operation as follows

$$K^{n+1} = K^n + \dot{K}^n \Delta t = 2K^n - K^{n-1}, \tag{4.27}$$

where $\dot{K}^n$ is the gradient of K at t_n. Using (4.26) and (4.27), (4.25) is rewritten as

$$u^{n+1} + \alpha \frac{\partial^2 u^{n+1}}{\partial y^2} = u^n + \beta \frac{\partial^2 u^n}{\partial y^2} + \frac{\Delta t}{2}(3K^n - K^{n-1}). \tag{4.28}$$

Thus equation (4.28) is the time discretization of the PDE (4.22) in which the terms on the RHS are determined from the previous steps. The first and second terms on the RHS are determined from a TPS-RBFN-based approximation of the current velocity field. The third term is obtained from a TPS-RBFN-based approximation of the data of the Brownian simulation process.

Specifically, to start the process, equation (4.28) is rewritten as follows

$$u^1 + \alpha \frac{\partial^2 u^1}{\partial y^2} = u^o + \beta \frac{\partial^2 u^o}{\partial y^2} + \frac{3K^o}{2}\Delta t, \tag{4.29}$$

where $K^o = \varsigma \frac{\partial \tau_p^{xy}}{\partial y}|_{t=o}$. It can be seen that the first term on the RHS of (4.29) is the initial condition of the problem.

In the present work, equation (4.28) with the boundary conditions (4.23) is solved for u at the time step $(n+1)$ using the linear least square principle. The sum of squared errors corresponding to the first step is given by

$$SSE(1) = \sum_{y_i \in \Omega} \left[\left(u^1(y_i) + \alpha \frac{\partial^2}{\partial y^2} u^1(y_i) \right) - \left(u^0(y_i) + \beta \frac{\partial^2}{\partial y^2} u^0(y_i) \right.\right.$$
$$\left.\left. + \frac{3}{2} \Delta t K^0(y_i) \right) \right]^2 + [u^0(0) - V]^2 + [u^0(L)]^2. \quad (4.30)$$

where Ω is the domain under consideration. u^1, τ^p_{xy} and their derivatives in equation (4.30) are approximated by either MQ-RBFNs (Eq (4.17), Eq (4.18), Eq (3.34) and Eq (4.19)) or TPS-RBFNs (Eq (4.17), Eq (4.18), Eq (3.37) and Eq (4.20)). Note that in the case of 1-D problem under consideration, y_i's are internal collocation points, $y_o = 0$ and $y_l = L$ are boundary collocation points. Generally, at time level $(n+1)$, the sum of squared errors is

$$SSE(n+1) = \sum_{y_i \in \Omega} \left[\left(u^{n+1}(y_i) + \alpha \frac{\partial^2}{\partial y^2} u^{n+1}(y_i) \right) - \left(u^n(y_i) + \beta \frac{\partial^2}{\partial y^2} u^n(y_i) \right.\right.$$
$$\left.\left. + \frac{\Delta t}{2} (3K^n(y_i) - K^{n-1}(y_i)) \right) \right]^2 + [u^n(0) - V]^2 + [u^n(L)]^2. \quad (4.31)$$

In the general case, the set of collocation points consists of the internal data and boundary data points. The collocation points can be distributed randomly or regularly. Here, the collocation points are arranged regularly and coincident with RBF centers. This choice gives the best results according to Kansa (1990a), Zerroukat *et al.* (1998) and Mai-Duy and Tran-Cong (2001). A system of linear algebraic equations is obtained in terms of the unknown weights by minimizing the SSE (4.30) and (4.31) and written as follows

$$\mathbf{G}\mathbf{w} = \mathbf{H}^T \hat{\mathbf{y}}, \quad (4.32)$$

where $\mathbf{G}$ is the variance matrix and described in §3.3.6 and Appendices B.2 and B.3. Here, each row of the design matrix $\mathbf{H}$ contains the values of the RBF corresponding to the terms $u^1(y_i) + \alpha \frac{\partial^2 u^1(y_i)}{\partial y^2}$; $\hat{\mathbf{y}}$ is a column vector whose elements correspond to the terms $u^0(y_i) + \beta \frac{\partial^2 u^0}{\partial y^2}(y_i) + \frac{3}{2}\Delta t K^o(y_i)$ and $\mathbf{w}$ is the vector of weights.

In this chapter, the unknown (weights) of the system of linear algebraic equation (4.32) is obtained by using the QR method (Dongarra *et al.*, 1979; Bjock, 1996). furthermore, for using MQ-RBFN based method, this method can produce the solution the solution at larger values of k in Eq (3.36) than the normal equation method arising from (4.32). The velocity field is thus described by the RBFNs once the sets of weights are calculated.

The numerical solution of the velocity field u^1 from equation (4.30) (in the least square sense) is the starting point for the solution of (4.31) at a general time step. The process continues until the steady state or a desired time is reached. The time discretisation is based on the Crank-Nicolson implicit method with $\theta = 0.5$ which reduces the total volume of calculation and is convergent and stable for a large range of $\frac{\Delta t}{\Delta y^2}$ (Smith, 1978; Carnahan *et al.*, 1969).

4.6 Algorithm of the CVFNNSS procedure

The general macro-microscopic approach mentioned in §4.3 can now be described in a more detailed algorithm as follows.

(1.) Start with a given initial velocity condition, generate a set of collocation points and an initial velocity field is approximated by RBF networks;

(2.) Generate an ensemble of homogenously distributed dumbbells over the flow domain. This initialisation of the polymer configuration field is based on the known equilibrium distribution function which is a three dimensional Gaussian distribution with zero mean and unit covariance (Öttinger, 1996; Bird *et al.*, 1987b);

(3.) Generate local volumes surrounding the collocation points. The local volumes are chosen such that each dumbbell is accounted for in one of these local volumes;

(4.) After the current velocity field is approximated by RBF networks, determine the velocity gradient field by calculating directly the gradient of the approximated velocity field for individual dumbbells;

(5.) Calculate the polymer configuration field (the connector vectors of the dumbbell ensemble) using the method described in §4.4. The velocity of the center of mass of each dumbbell is considered to be equal to the pointwise local fluid velocity;

(6.) Determine the local stress tensor by taking the ensemble average of the polymer configuration on each local volume and assign this stress to the collocation point associated with this volume. The stress is then approximated globally on the whole domain by RBF networks which are the ultimate description of the stress field. This global approximation procedure smooths the piecewise continuous stress field with a globally continuous function. This could be achieved by either TPS-RBFNs or MQ-RBFNs. However, the former was proved to have superior smoothing charateristic (Beatson and Light, 1997) and hence is used in the present work;

(7.) With the stress field just obtained, solve the set of conservation equations for the new velocity field using a mesh-free RBFN method as described in §4.5;

(8.) Terminate the simulation when either the desired time or steady state is reached. The latter is determined by a convergence measure for either the velocity field or the stress field between two consecutive iterations which is defined for velocity field by

$$CM = \sqrt{\frac{\sum_1^N \sum_{i=1}^d (u_i^n - u_i^{n-1})^2}{\sum_1^N \sum_{i=1}^d (u_i^n)^2}} < tol, \tag{4.33}$$

where d is the number of dimension (1 in the present work); tol is a preset tolerance; u_i is the i component of the velocity at a node; N is the total number of collocation points and n is the iteration number. Convergence is also checked for the shear stress and the first normal stress difference;

(9.) Return to step (4) for the next time level.

4.7 Numerical examples

The aim of this section is to report an initial assessment of the validity and efficiency of the present meshless method and therefore the start-up planar Couette flow is considered using two kinds of kinetic dumbbell models: the FENE and FENE-P. This problem, already described in §4.5.2 and Fig. 4.2, was solved using the FENE-P model by Mochimaru (1983), and FENE/FENE-P by Laso and Öttinger (1993) while Fan (1985) provided a steady state solution using the FENE model. In order to compare the present results with those from Mochimaru (1983), Laso and Öttinger (1993), the same number of dumbbells $M = 50000$ is homogeneously distributed into the domain and the same non-dimensionlisation scheme as in Laso and Öttinger (1993) is used. However, before studying these models, the reliability of the present method is also investigated using Hookean model (Oldroyd-B) since the corresponding velocity field is known in closed form. The non-interacting dumbbells are neutrally suspended in a Newtonian solvent of known viscosity η_s, density ϱ and the resultant material is characterized by the relaxation time λ_H. The dimensionless gap $L = 1$, moving plate velocity $V = 1$ and zero-shear-rate viscosity $\eta_o = \eta_s + \eta_p = 1$. A set of N collocation points is generated in the volume either randomly or regularly. In order to compare with other methods, the same local volumes are chosen, namely line segments of equal length $L/(N-1)$ except the two volumes near the boundaries which are $L/2(N-1)$. The convergence measure is set at $1.0e-4$ and the simulation is continued for $t \geq 0$ until the flow reaches the steady state.

4.7.1 Start-up planar Couette flow with the Hookean model

The Hookean (Oldroyd-B) model has a closed-form constitutive equation and the start-up planar Couette flow of this fluid can be found by inverting the Laplace transform as follows (Laso and Öttinger, 1993)

$$L\left\{u_x\left(y,t\right)\right\} = \frac{1}{s}\frac{\sinh\sqrt{\frac{\rho s(1+\lambda_1 s)}{\eta_o(1+\lambda_2 s)}}\left(1-y\right)}{\sinh\sqrt{\frac{\rho s(1-\lambda_1 s)}{\eta(1-\lambda_2 s)}}}$$

where ρ is density of the fluid, $\eta_o = \eta_p + \eta_s$, $\eta_p = nkT\lambda_H$; λ_1 is the relaxation time ($\lambda_1 = \lambda_H$), λ_2 is the retardation time ($\lambda_2 = \frac{\lambda_H \eta_s}{\eta_o}$); $s(t)$ is time. In this example, the parameters are given as follows: $\rho = 1.2757$; $\eta_s = 0.0521$; $\alpha = \frac{\eta_s}{\eta_o} = 0.0521$,$\lambda_H = 49.62$. Typical global ensemble size of 400000 dumbbells and time step of $\Delta t = 10e-2$ are used. While analytical result of the velocity is determined using 31 collocation points, the results of velocity, computed from the present method, are obtained using two different numbers of collocation points $N = 31$, and 26. Figure 4.3 depicts the evolution of the velocity profile at four location $y = 0.2$, 0.4, 0.6 and 0.8 using the analytical method (solid line), and the present method (dashed lines). This figure shows that results of the present method are in good agreement with those of the analytical method although the present method is a stochastic process in nature.

4.7.2 Start-up planar Couette flow with the FENE model

The FENE model has no closed-form constitutive equation and the problem was solved recently by Laso and Öttinger (1993) using the CONNFFESSIT approach. In this work, as in Laso and Öttinger (1993), the parameters are: 50000 dumbbells, 41 collocation points, $\rho = 1.2757$, $\lambda_H = 49.62$, $b = 50$, $\eta_s = 0.0521$, $\Delta t = 10^{-2}$ ($\Delta t = 10^{-4}$ in Laso and Öttinger (1993)). The numerical solution by the present method confirms the velocity overshoot of viscoelas-

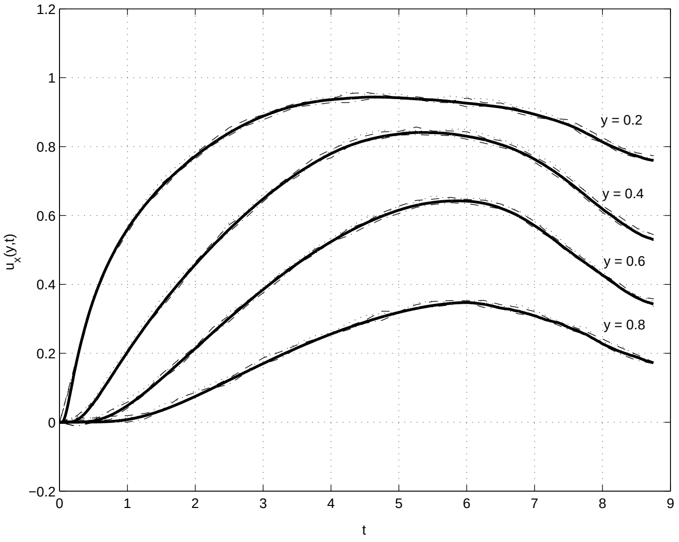


Figure 4.3: The start-up planar Couette flow problem using the Hookean dumbbell model: the parameters of the problem are number of dumbbells $M = 50000$, $\lambda_H = 49.62$, $\eta_s = 0.521$, and $\Delta t = 10^{-2}$. The time evolution of the velocity at $y = 0.2$, $y = 0.4$, $y = 0.6$, $y = 0.8$. The solid line represents the analytical solution, given in Laso and Öttinger (1993), using $N = 31$, and the dashed lines represent the solution of the present CVFNNS method using different numbers of collocation points $N = 31$ and $N = 26$.

tic fluids and is in complete agreement with the findings of Laso and Öttinger (1993).

Figures 4.4 and 4.5 describe the evolution of the velocity profile, which shows that velocity undershoot is insignificant in comparison with overshoot. Fig. 4.6 shows that typical time evolutions of the velocity at four locations $y = 0.2$, $y = 0.4$, $y = 0.6$ and $y = 0.8$ do not differ significantly for the cases $N = 41$ and 31, indicating that $N = 31$ is an adequate number of collocation points. Fig. 4.6 also shows that velocity overshoot occurs sooner in fluid layers nearer to the moving wall. The convergence behavior of velocity field with respect to the iteration number for the case $N = 21$, 26, 31 and $N = 41$, Fig. 4.7, shows that the method is convergent even with a coarse density of collocation points. Figures 4.8 and 4.9 show a typical evolution of the shear stress and the first normal stress difference, respectively, at locations $y = 0.8$ for the cases

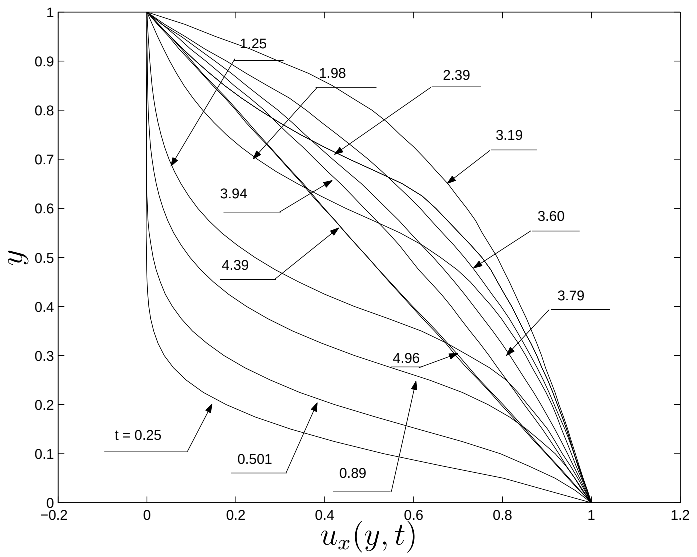


Figure 4.4: The start-up planar Couette flow problem using the FENE dumbbell model: the parameters of the problem are number of dumbbells $M = 50000$, number of collocation points $N = 41$, $\lambda_H = 49.62$, $b = 50$, $\eta_s = 0.0521$ and $\Delta t = 10^{-2}$. The velocity profile with respect to location y at different times shows velocity overshoot but hardly any oscillation.

of $N = 41$, 31, 26 and 21 collocation points. Although the figures may present interesting time-dependent behaviour of the flow, a rheological explanation is not within the primary numerical scope of this work. On the other hand, Fig. 4.10 and Fig. 4.11 depict the evolution of shear stress and the first normal stress difference, respectively, at locations $y = 0.2$, $y = 0.4$, $y = 0.6$ and $y = 0.8$ using $N = 31$. The stress response is sharper near the moving wall which is consistent with the velocity overshoot behaviour. The time-step size Δt influences the accuracy of the microscopic stochastic integration. Generally, the larger is the time-step size, the bigger is the mean error of solution of SDE's (4.9) or (4.11). However, when Δt is very small the variance may be large due to roundoff errors and can destroy the result (Kloeden et al., 1997, Kloeden and Platen, 1997). Furthermore, a very good agreement with the results of other methods is obtained by the present method with a coarse set of collocation points. Fig. 4.12 shows the shear stress profiles τ_{yx} and the statistical error bars with respect to location y for a sampling of 120 computations using two

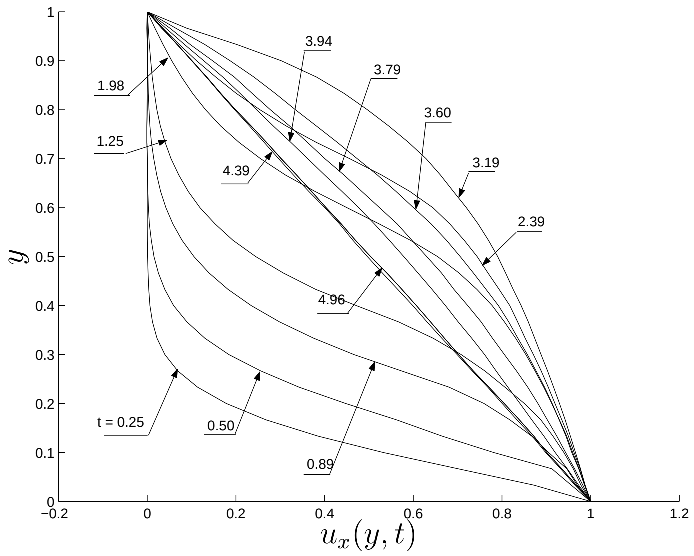


Figure 4.5: The start-up planar Couette flow problem using the FENE dumbbell model: the parameters are same as in Fig. 4.4 except that the number of collocation points is less ($N = 31$).

time-step sizes, $\Delta t = 10^{-2}$ and 10^{-3}, and 26 collocation points, at the following times $t = 0.60, 3.89, 5.82, 15.00$ and 35.00. The results show that the statistical errors are small and stable at the steady state.

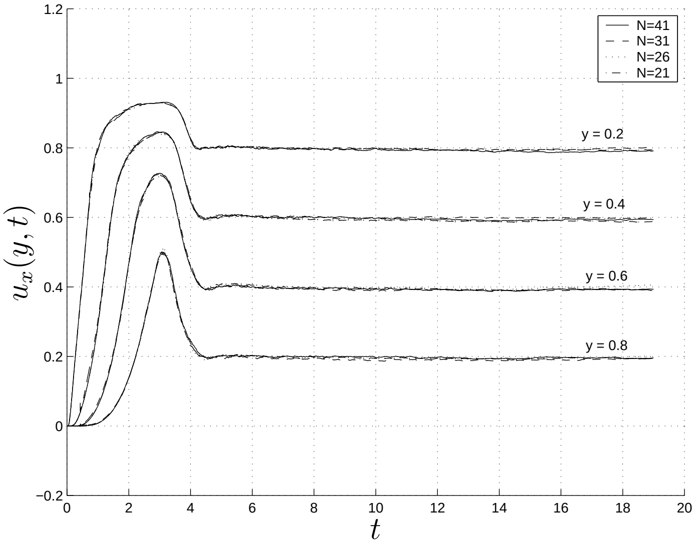


Figure 4.6: The start-up planar Couette flow problem using the FENE dumbbell model: the parameters are same as in Fig. 4.4. The influence of the number of collocation points on the time evolution of the velocity at locations $y = 0.2$, $y = 0.4$, $y = 0.6$ and $y = 0.8$.

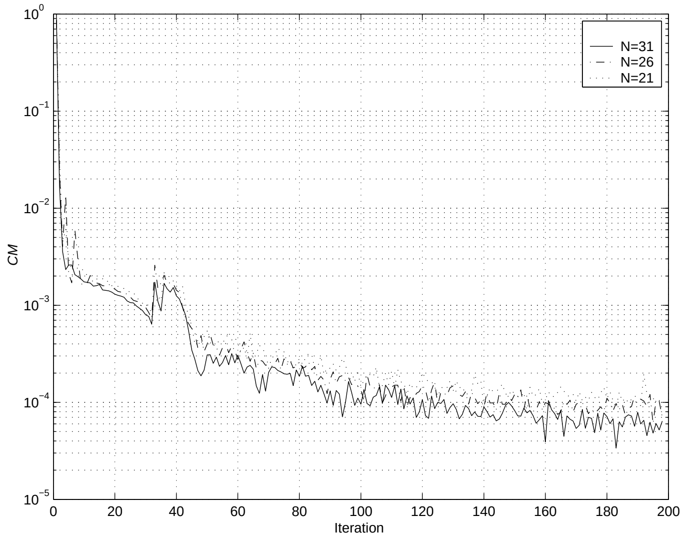


Figure 4.7: The start-up planar Couette flows using the FENE model in the present CVFNNSS method: the convergence behavior (CM) of the velocity field with respect to the iteration number for $N = 31$, 26 and 21 collocation points

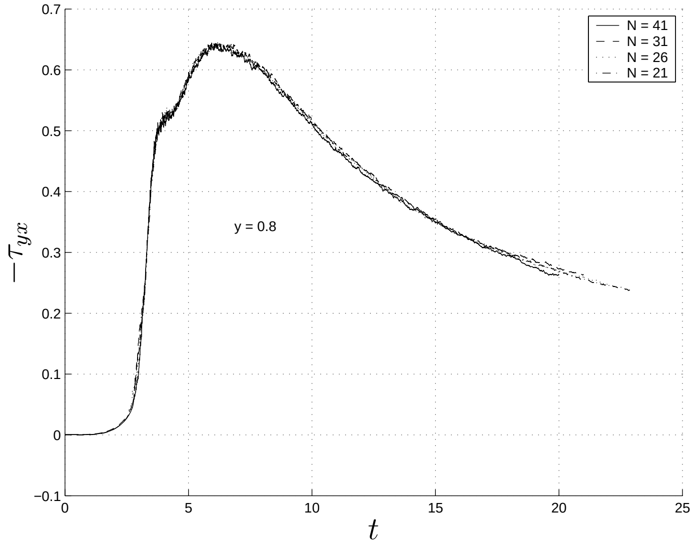


Figure 4.8: The start-up planar Couette flows using the FENE model: the parameters other are shown in Fig. 4.4. The evolution of shear stress at the location $y = 0.8$ with respect to time for $N = 41$, 31, 26 and 21.

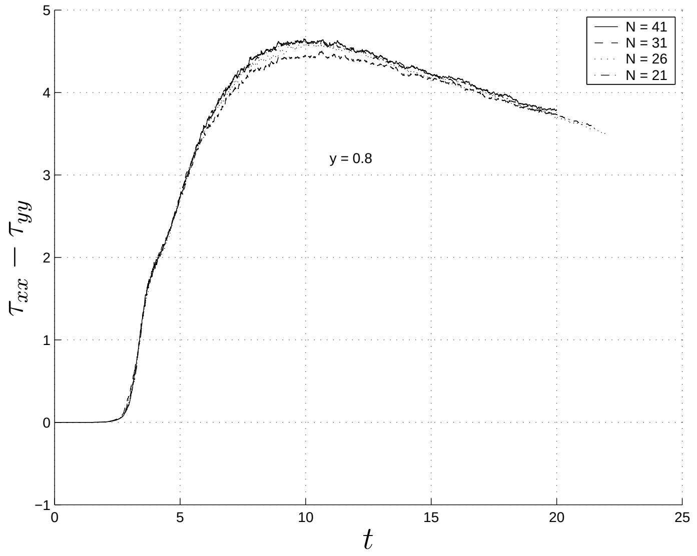


Figure 4.9: The start-up planar Couette flows using the FENE model: the parameters other are the same as shown in Fig. 4.4. The evolution of the first normal stress difference at location $y = 0.8$ with respect to time for $N = 41$, 31, 26 and 21.

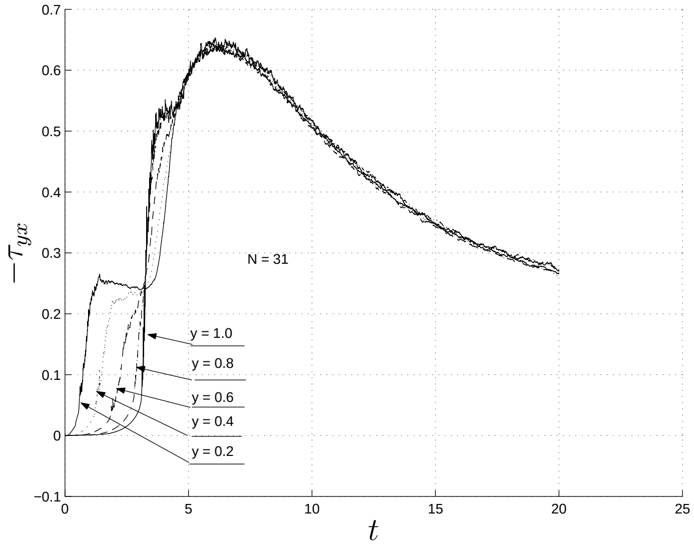


Figure 4.10: The start-up planar Couette flows using the FENE model: the parameters are the same as shown in Fig. 4.4 except that the number of collocation points is 31. The evolution of shear stress at locations $y = 0.2, y = 0.4, y = 0.6, y = 0.8$ and $y = 1.0$ with respect to time.

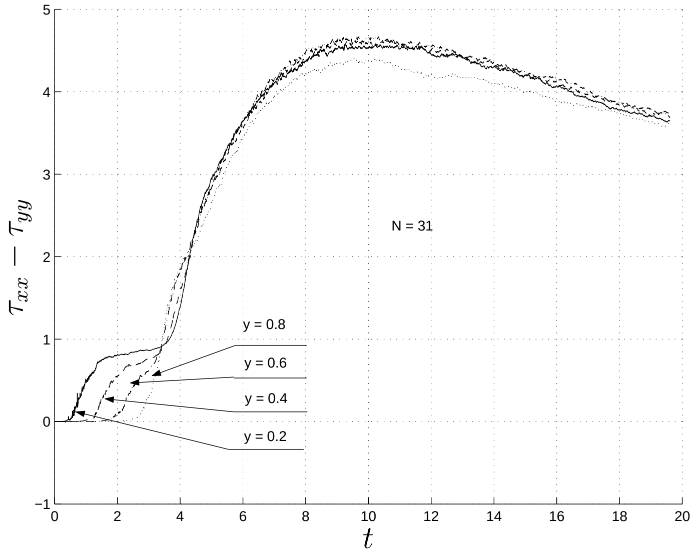


Figure 4.11: The start-up planar Couette flows using the FENE model in the present CVFNNSS method: the parameters are the same as shown in Fig. 4.4 except that the number of collocation points is decreased from 41 to 31. The evolution of the first normal stress differences at locations $y = 0.2, y = 0.4, y = 0.6$ and $y = 0.8$ with respect to time.

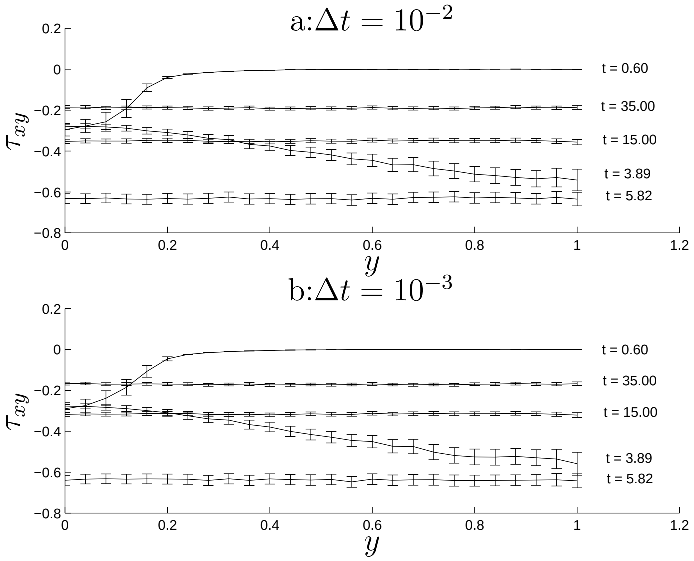


Figure 4.12: The start-up planar Couette flows using the FENE model in the present CVFNNSS method: the shear stress profiles τ_{yx} and the statistical error bar with respect to location y at different times $t = 0.60, 3.89, 5.82, 15.00$ and 35.00: the parameters are the same as shown in Fig. 4.4 except that the number of collocation points is equal to 26 for both time-step sizes (a) $\Delta t = 10^{-2}$ and (b) $\Delta t = 10^{-3}$.

4.7.3 Start-up planar Couette flow with the FENE-P model

The start-up planar Couette flow problem using the FENE-P model was solved by Mochimaru (1983) where the macroscopic numerical approach made use of the constitutive equation derived from the kinetic theory of a dilute solution of the FENE-P dumbbells in a Newtonian fluid (e.g. equation (13.5-56) of Bird *et al.* (1987b)). The problem with the same material parameters was also solved by Laso and Öttinger (1993) using the CONNFFESSIT method. Similarly, the problem is solved by the present CVFNNSS method with the same parameters as in Mochimaru (1983), Laso and Öttinger (1993), i.e. $\rho = 1.2325, \quad \lambda_H = 49.62, \quad b = 50, \quad \eta_s = 0.050332$. However the time increment Δt in the present method can be as high as 10^{-2} which appears to be an improvement in comparison with Laso and Öttinger (1993) where $\Delta t = 1.0e-$. Figs. 4.13-4.15 show the evolution of velocity field which exhibits oscillatory transient behaviour. Keeping the number and distribution of dumbbells the same, the number of collocation points is varied from 21 to 41 (Figs. 4.15-4.17) to confirm that the obtained results are accurate. The present method are in excellent agreement with the theoretical results by Mochimaru (1983), Laso and Öttinger (1993) and the experimental results of Chow and Fuller (1985) confirm the velocity overshoots (Fig. 4.13-4.15) and stress propagation between the plates (Fig. 4.18-4.19).

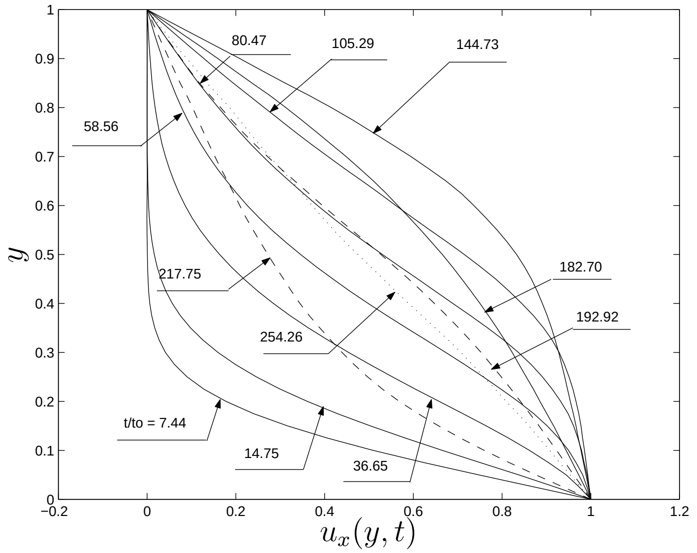


Figure 4.13: The start-up planar Couette flow problem using the FENE-P dumbbell model in the present CVFNNSS method: the parameters are number of dumbbells $M = 50000$, number of collocation points $N = 41$, $\lambda_H = 49.62$, $b = 50$, $\eta_s = 0.0521$ and $\Delta t = 10^{-2}$. The velocity profile with respect to location y at different times shows the strong velocity overshoots and undershoots. In this plot the time is non-dimensionalised with respect to the reference time t_o as defined in Mochimaru (1983) for easy comparison with the latter results.

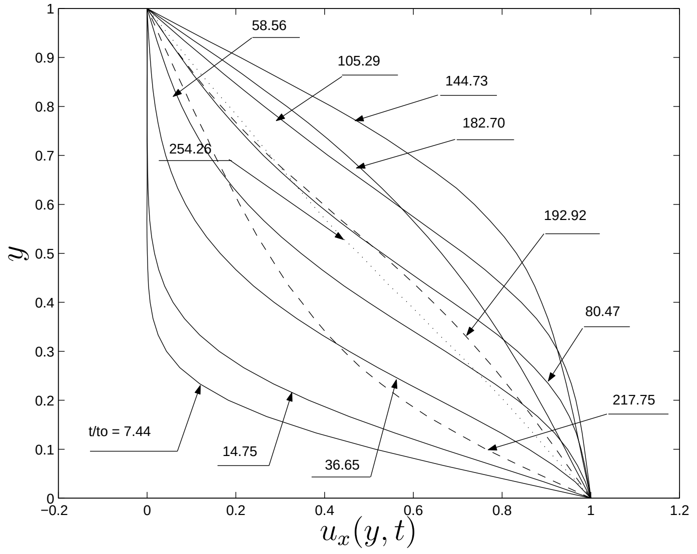


Figure 4.14: The start-up planar Couette flow problem using the FENE-P dumbbell model in the present CVFNNSS method: the parameters are the same as shown in Fig. 4.13 except that the number of collocation points is decreased from 41 to 31. The results here show no significant difference from those of Fig. 4.13.

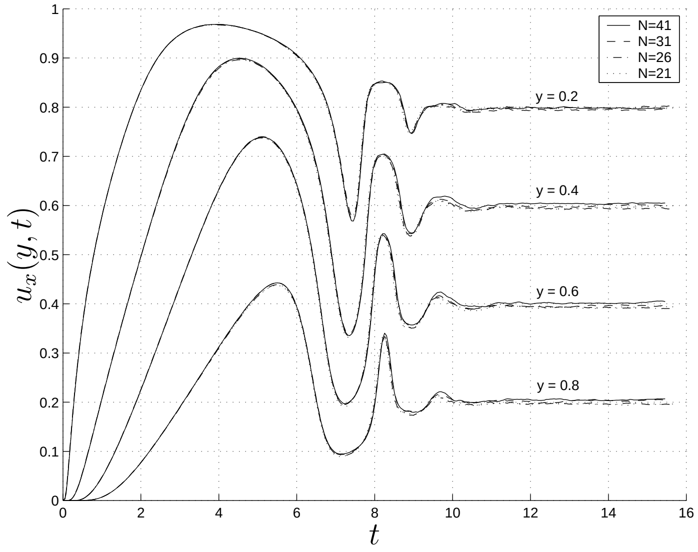


Figure 4.15: The start-up planar Couette flow problem using FENE-P dumbbell model in the present CVFNNSS method: the parameters other than N are shown in Fig. 4.13. The evolution of the velocity field with respect to time at locations $y = 0.2$, $y = 0.4$, $y = 0.6$ and $y = 0.8$ for the cases of $N = 41$, 31, 26 and 21.

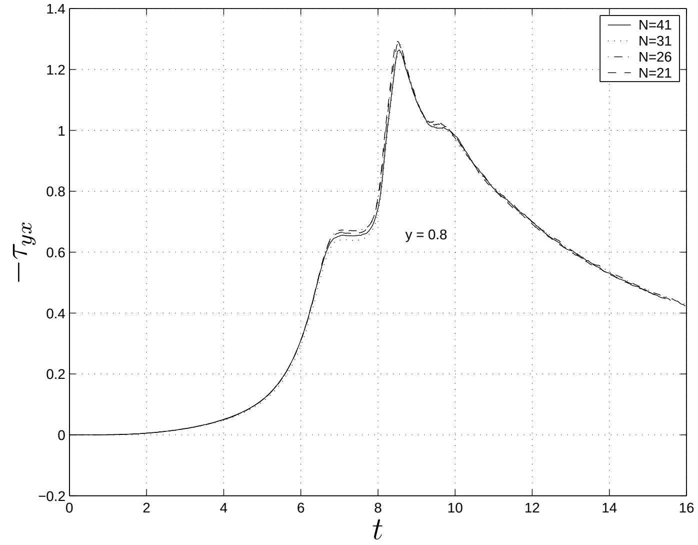


Figure 4.16: The start-up planar Couette flows using the FENE-P model in the present CVFNNSS method: the parameters other than N are shown in Fig. 4.13. The evolution the shear stress at the location $y = 0.8$ with respect to time for the cases of $N = 41$, 31, 26 and 21.

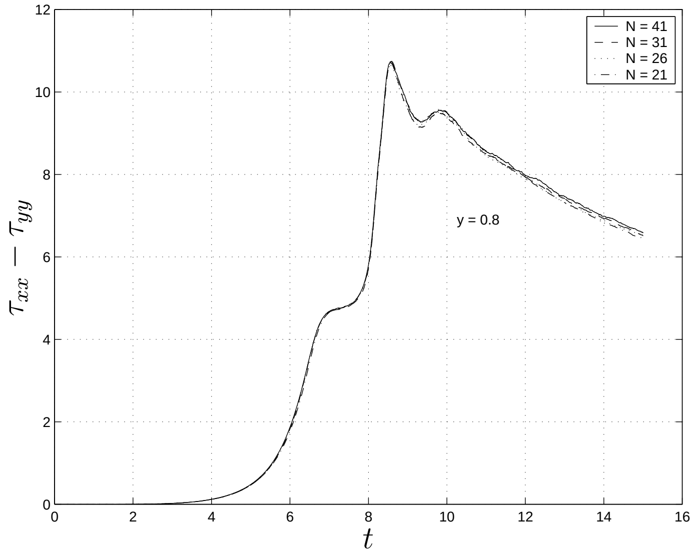


Figure 4.17: The start-up planar Couette flows using the FENE-P model in the present CVFNNSS method: the parameters other than N are the same as shown in Fig. 4.13. The evolution of the first normal stress difference at location $y = 0.8$ with respect to time for the cases of $N = 41$, 31, 26 and 21.

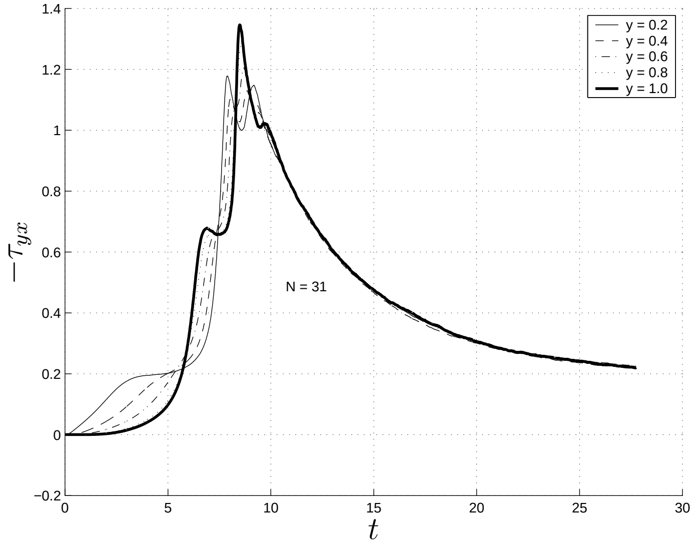


Figure 4.18: The start-up planar Couette flows using the FENE-P model in the present CVFNNSS method: The parameters are the same as shown in Fig. 4.13 except that the number of collocation points is decreased from 41 to 31. The evolution of the shear stresses at locations $y = 0.2, y = 0.4, y = 0.6, y = 0.8$ and $y = 1.0$ with respect to time.

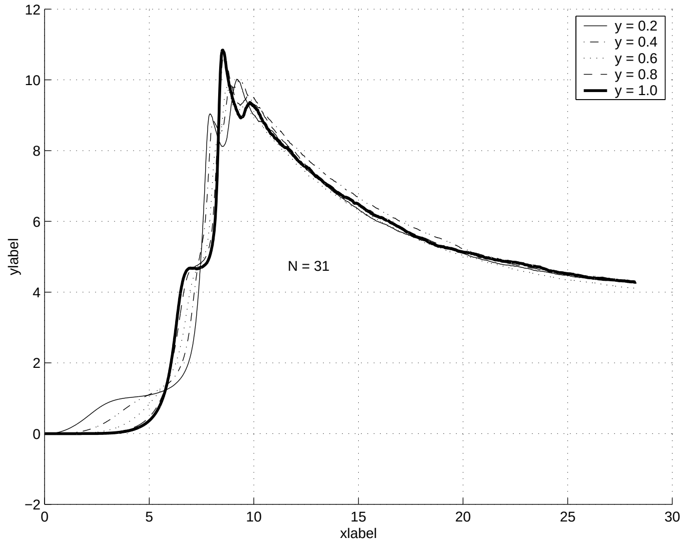


Figure 4.19: The start-up planar Couette flows using the FENE-P model in the present CVFNNSS method: the parameters are the same as shown in Fig. 4.13 except that the number of collocation points is decreased from 41 to 31. The evolution of the first normal stress differences at locations $y = 0.2, y = 0.4, y = 0.6, y = 0.8$ and $y = 1.0$ with respect to time.

4.7.4 Comparison between the FENE and FENE-P models

The comparison between the FENE and FENE-P models in planar Couette start-up flows is summarised in Figs. 4.20-4.22. Fig. 4.20 denotes the time development of the velocity field at four locations $y = 0.2 \quad y = 0.4 \quad y = 0.6$ and $y = 0.8$ between the FENE and FENE-P models. It shows that there is a big difference in dynamic responses of the two models, but the difference is non-significant after the flow reaches the steady state. Fig.4.20 also shows that the duration of the velocity overshoot is much longer for the FENE-P model and the steady state seems to take much longer to be reached than for the FENE model. The strong oscillatory behaviour of the velocity and stress fields due to the linearisation of the FENE-P models is in excellent agreement with results presented in Fig. 5 and 6 of Laso and Öttinger (1993).

Figure 4.21 depicts the difference of the evolution of shear stresses between the FENE and FENE-P models at the fixed plate with respect to time. It can be seen that the FENE-P model produces a maximum of the shear stress about twice the corresponding value for FENE, however the maximum value seems to take longer to be reached for the FENE-P model than for the FENE model. Furthermore, the asymtotic values of the stress are the same at the steady state in agreement with Laso and Öttinger (1993) and Herrchen and Öttinger (1997) and the steady state is reached in about the same time.

The comparison of the first normal stress difference between the FENE and FENE-P models at the fixed plate is shown in Fig. 4.22 where the FENE-P model shows a secondary peak at time 10. This physical peak obtained from the present method is in agreement with results obtained from macroscopic and mesoscopic methods (see Fig.6 and Fig. 7 in Laso and Öttinger, 1993). Figs. 4.23 and 4.24 show the influence of noise on the shear stress and first normal stress difference with respect to time at the fixed plate using 50 different random shifts. The maximum values of the first normal stress difference seem to take the same time to be reached for both models, despite the delay in the wall shear stress maximum in the case of the FENE-P model. However, the values at the steady state are different although they reach the steady state at about the same time.

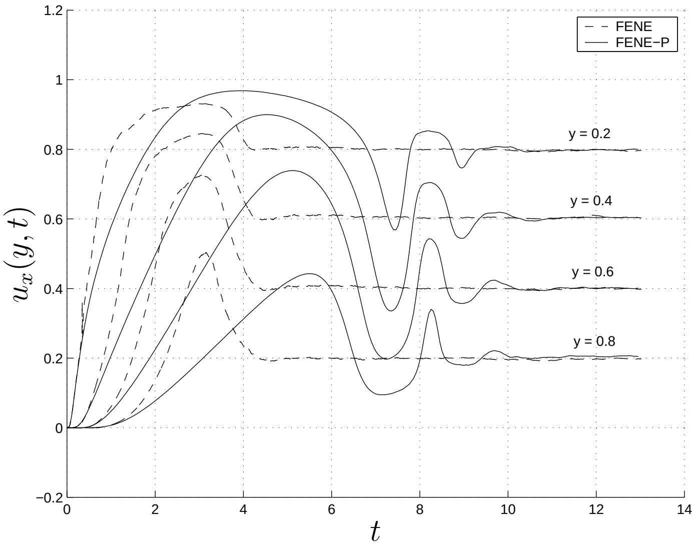


Figure 4.20: The start-up planar Couette flows using FENE and FENE-P models in the present CVFNNSS method: the parameters are shown in Fig. 4.4 for the FENE model and Fig. 4.13 for the FENE-P model. Comparison of the velocity fields with respect to time at locations: $y = 0.2$, $y = 0.4$, $y = 0.6$ and $y = 0.8$.

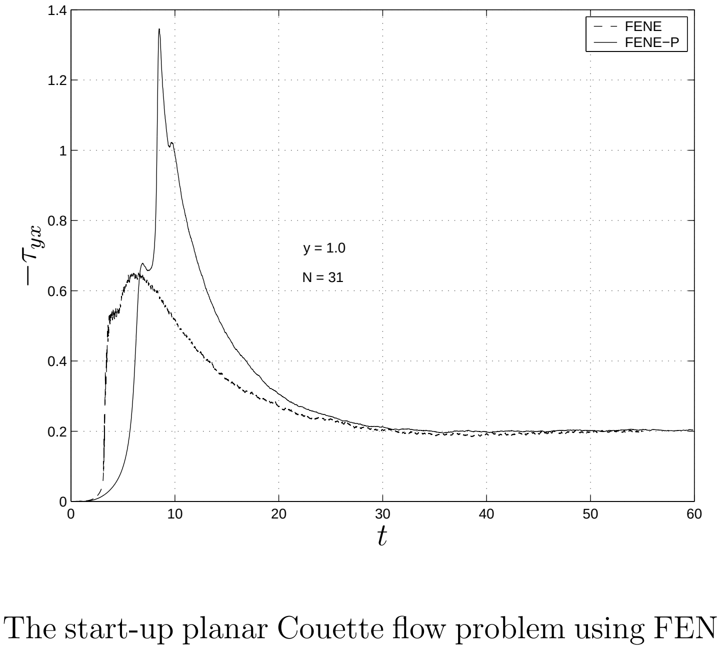


Figure 4.21: The start-up planar Couette flow problem using FENE and FENE-P models in the present CVFNNSS method: the parameters are shown in Fig. 4.4 for the FENE model and Fig. 4.13 for the FENE-P model. Comparison of the shear stress at the fixed plate with respect to time between the two models.

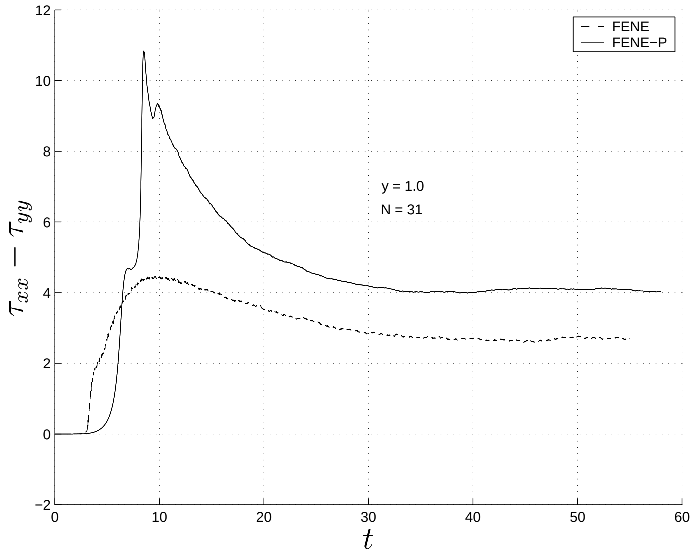


Figure 4.22: The start-up planar Couette flow problem using the FENE and FENE-P models in the present CVFNNSS method: the parameters are shown in Fig. 4.4 for the FENE model and Fig. 4.13 for the FENE-P model. Comparison of the first normal stress difference at the fixed plate with respect to time between the two models.

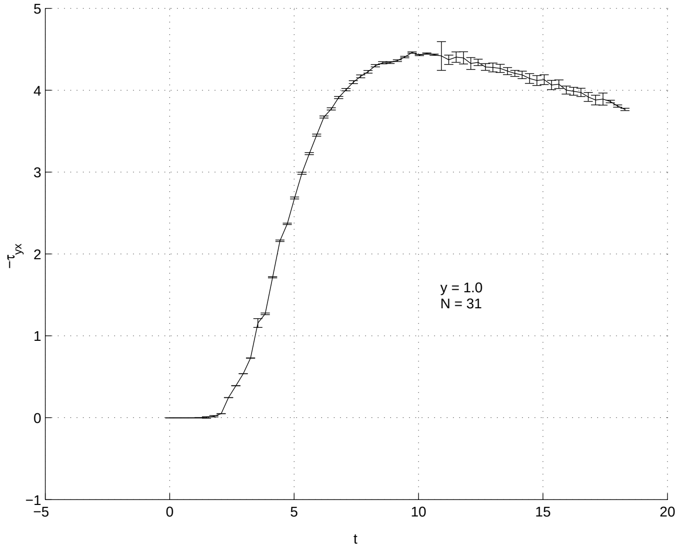


Figure 4.23: The start-up planar Couette flow problem using the FENE model in the present CVFNNSS method: the parameters are shown in 4.4. The influence of Brownian noise on the shear stress with respect to time at the fixed plate, using 50 different random shifts.

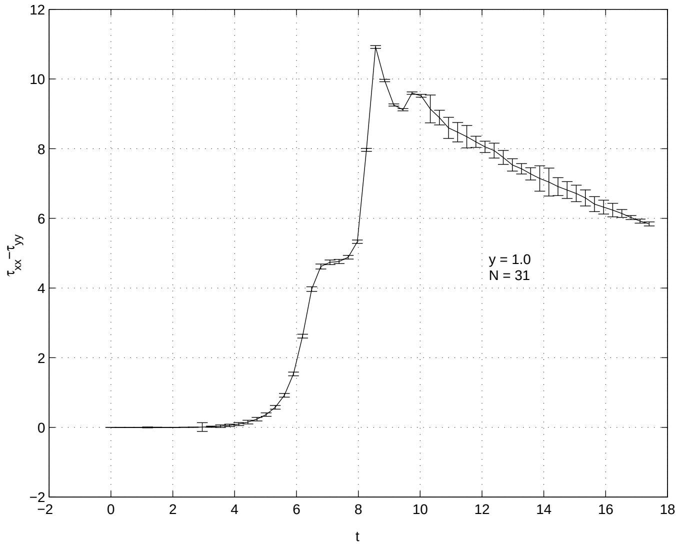


Figure 4.24: The start-up planar Couette flow problem using the FENE-P model in the present CVFNNSS method: the parameters are shown in 4.13. The influence of Brownian noise on the first normal stress difference with respect to time at the fixed plate, using 50 different random shifts

4.8 Concluding remarks

This chapter has demonstrated a successful adaptation of the macro-micro approach introduced in the CONNFFESSIT method. The finite element-based macro-procedure of the CONNFFESSIT is replaced by the present meshless neural network-based procedure. For the start-up Couette flow with the FENE and FENE-P models, a complete agreement on the typical flow features with the results of Mochimaru (1983) and Laso and Öttinger (1993) is obtained. The present CVFNNSS method retains the properties inherent in the CONNFFESSIT (Öttinger, 1996; Laso and Öttinger, 1993; Feigl *et al.*, 1995; Laso *et al.*, 1997, 1999) namely (i) easy handling of complex polymer models without closed form constitutive equation, (ii) easy switching between different models, (iii) realistic treatment of boundary conditions. Furthermore, the present CVFNNSS has the advantage of being a mesh-free numerical method where the domain discretization for the governing PDE's is simply an unstructured set of collocation points. Owing to the approximation characteristics of RBFNs, the initial conditions are represented in a more natural way. For shear flows, the present CVFNNSS method appears to be much more stable than other methods reviewed here and becomes unstable only at $\Delta t = 5e-2$. Furthermore, with a coarse set of collocation points, the present method gives results with a similar accuracy in comparison with those from other schemes. However, these initial results are to be confirmed with more complex flows (e.g. elongational flows) in the next stage of investigation where numerical issues such as stochastic integration schemes and choice of collocation points will be of major concern. The noise arising in the velocity field and specially in the stress tensor (Fig. 4.8-4.11, Fig. 4.16-4.19 and Fig. 4.21-4.22) due to the Brownian motion can be drastically reduced by variance reduction methods (Öttinger *et al.*, 1997; Bonvin and Picasso, 1999). Variance reduction methods will also be taken into account when the present method is implemented for higher dimensional problems and its ability is tested in the next chapter.

Chapter 5

Element-free computation of 2D viscoelastic flows using CVFNNSS

The CVFNNSS method (Tran-Canh and Tran-Cong, 2002b), which was described and verified in the previous chapter for 1D problems, is further developed for 2D steady-state problems in this chapter. The planar Poiseuille flow is used to verify the method and the 4:1 axisymmetric contraction flow is simulated for materials characterized by Hookean, FENE and FENE-P dumbbell models.

5.1 Introduction

The proposed method is based on the combination of RBFNs and Lagrangian hybrid simulation using SST, in which the stress is computed from the Brownian dynamics simulation through the particle tracking, and the velocity field is determined by solving the equations of mass and momentum conservation with a mesh-free method based on RBFNs which does not require any kind of finite element-type discretisation of the domain and its boundary. The variance reduction technique, which is based on a control variate scheme, is employed for the Brownian dynamics simulations. This chapter describes additional details of the microscopic component and then concentrates on the macroscopic component of the computational method for 2D viscoelastic flows. The chapter is organized as follows. In §5.2 and §5.3, an outline of the governing equations and the stochastic simulation technique for computing the stress are presented. In §5.4 the numerical methods for solving the SDEs and the conservation equations and the control variate variance reduction technique are described. §5.5 presents the algorithm of the present scheme for 2D viscoelastic problems highlighting the macroscopic-microscopic interfaces of the method. The numerical examples are then discussed in §5.6 with a brief conclusion in §5.7.

5.2 Governing Equations for steady viscoelastic flow problems

Considering the isothermal and steady flow of an incompressible fluid, the system of momentum and mass conservation equations is given by

$$\nabla \cdot \mathbf{u} = 0, \tag{5.1}$$

$$-\nabla p + \boldsymbol{\nabla} \cdot \boldsymbol{\tau} = \rho(\mathbf{u} \cdot \nabla)\mathbf{u}, \tag{5.2}$$

where $\mathbf{u}$ denotes the velocity field; $\boldsymbol{\tau}$ is the extra stress; ρ is the fluid density. For a polymer solution, the extra stress $\boldsymbol{\tau}$ can be decomposed as

$$\boldsymbol{\tau} = 2\eta_N \mathbf{D} + \boldsymbol{\tau}^p, \tag{5.3}$$

where $2\eta_N\mathbf{D}$ is the Newtonian solvent contribution; η_N is the solvent viscosity; $\mathbf{D} = \frac{1}{2}\left(\nabla\mathbf{u} + (\nabla\mathbf{u})^T\right)$ is the rate of strain tensor; $\boldsymbol{\tau}^p$ is the polymer-contributed stress and p is the pressure arisen from the incompressibility constraint. Using

Eq (5.3), Eq (5.2) can be rewritten as follows

$$2\eta_N \boldsymbol{\nabla} \cdot \mathbf{D} - \rho(\mathbf{u} \cdot \boldsymbol{\nabla})\mathbf{u} - \boldsymbol{\nabla} p = -\nabla \cdot \boldsymbol{\tau}^p. \tag{5.4}$$

Instead of using the continuity equation (5.1), the incompressibility condition is enforced via the penalty method as follows (Bernstein *et al.*, 1994 and Laso *et al.*, 1999)

$$p = -p_e(\boldsymbol{\nabla} \cdot \mathbf{u}), \tag{5.5}$$

where p_e is a sufficiently large penalty parameter. Thus, the momentum equation (5.4) is rewritten as

$$2\eta_N \boldsymbol{\nabla} \cdot \mathbf{D} - \rho(\mathbf{u} \cdot \boldsymbol{\nabla})\mathbf{u} + p_e \boldsymbol{\nabla}(\boldsymbol{\nabla} \cdot \mathbf{u}) = -\boldsymbol{\nabla} \cdot \boldsymbol{\tau}^p. \tag{5.6}$$

Although this method produces an error of $O(p_e^{-1})$ (Baker, 1983) in approximating $\nabla \cdot \mathbf{u} = 0$, it is considered as a good method which allows the elimination of the incompressibility condition and a corresponding reduction of the number of degrees of freedom of the problem in solving complex problems (Hughes *et al.*, 1979; Crochet *et al.*, 1984; Bernstein *et al.*, 1994; Laso *et al.*, 1997, 1999). Travis *et al* (1990) have made rigorous comparison between a number of numerical methods and concluded that the methods based on penalty function produce comparably accurate results. The value of the penalty parameter can only be chosen from experience at this stage and the value chosen in this work is based on the results reported in the references cited above.

The polymer-contributed stress $\boldsymbol{\tau}^p$ is determined numerically via a microscopic technique which is based on the BDS and kinetic modelling. In the next subsection, the Lagrangian SST is described.

5.3 Review of Lagrangian SST in polymeric kinetic theory

5.3.1 Governing SDE for a dumbbell

From presentation given in chapter 1, instead of solving the diffusion equation directly, the polymer configuration $\mathbf{Q}(t)$ is determined by using an equivalent SDE. The models, which consist of elastic dumbbells having two Brownian beads attached by an entropic spring, satisfy the SDE as follows

$$d\mathbf{Q} = \left[\boldsymbol{\kappa} \cdot \mathbf{Q} - \frac{2}{\zeta}\mathbf{F}(\mathbf{Q})\right] dt + \sqrt{\frac{4k_B T}{\zeta}} d\mathbf{W}(t), \tag{5.7}$$

where ζ is the friction coefficient between the dumbbell and the solvent; $\mathbf{W}(t)$ is a 3-component column vector which is a Wiener process with mean $\langle W_i(t)\rangle = 0$ and covariance $\langle W_i(t)W_j(t')\rangle = \delta_{ij}\min(t,t')$ which accounts for the random displacement of the beads due to thermal motion; $\mathbf{F}$ is the spring connector force between the two beads; $\boldsymbol{\kappa}$ is the velocity gradient. Let H be the spring constant, the connector force is given by

$$\mathbf{F} = H\mathbf{Q}, \quad \mathbf{F} = H\frac{\mathbf{Q}}{1-\left(\frac{\mathbf{Q}}{Q_o}\right)^2}, \quad \mathbf{F} = H\frac{\mathbf{Q}}{1-\frac{\langle \mathbf{Q}^2\rangle}{Q_o^2}}, \tag{5.8}$$

for the Hookean, FENE and FENE-P dumbbell models, respectively, where Q_o is the maximum possible spring length.

It can be considered that molecules are convected by macroscopic fluid flows where the centres of mass of the molecules move along trajectories which satisfy the following equation

$$\frac{dx_i}{dt} = v_i(t, \mathbf{x}), \tag{5.9}$$

where v_i is the i-component of the velocity of a dumbbell whose coordinate is $\mathbf{x}$. Please see §4.4 for more details.

5.3.2 Non-dimensionalisation

Let $\lambda_H = \frac{\zeta}{4H}$ be a characteristic relaxation time and $b = \frac{HQ_o^2}{k_BT}$ is the square of the dimensionless maximum extensibility. The η_N is scaled by η_o ($\eta_o = \eta_N + \eta_p$), $\mathbf{Q}$ by $\sqrt{\frac{k_BT}{H}}$, polymer stress by $n_b k_B T$, the gradient $\boldsymbol{\kappa}$ by λ_H^{-1} and time t by λ_H. The dimensionless variables are given by

$$\eta_N^{'} = \frac{\eta_N}{\eta}; \qquad \mathbf{Q}^{'} = \frac{\mathbf{Q}}{\sqrt{\frac{k_BT}{H}}}; \qquad t^{'} = \frac{t}{\lambda_H}; \qquad \boldsymbol{\kappa}^{'} = \boldsymbol{\kappa}\lambda_H; \qquad \boldsymbol{\tau}_p^{'} = \frac{\boldsymbol{\tau}_p}{n_d k_B T},$$

where n_d is the number of dumbbells per unit volume of the solution. For the sake of brevity, the dimensionless forms are written without prime from here on.

It is noted for further reference that the polymer contribution to the zero shear rate viscosity is $\eta_{p(\text{Hookean})} = n_d k_B T \lambda_H$, $\eta_{p(\text{FENE})} = n_d k_B T \lambda_H b/(b+5)$ and $\eta_{p(\text{FENE-P})} = n_d k_B T \lambda_H b/(b+3)$ (see Appendix C.2 for details).

The dimensionless form of the SDE (5.7) is now written as

$$d\mathbf{Q} = \left[\boldsymbol{\kappa}\cdot\mathbf{Q} - \frac{\mathbf{F}}{2}\right] dt + d\mathbf{W}(t), \tag{5.10}$$

where $\mathbf{F}^{'}$ is the dimensionless spring force and is given by

$$\mathbf{F} = \mathbf{Q}, \quad \mathbf{F} = \frac{\mathbf{Q}}{1 - \frac{\mathbf{Q}^2}{b}}, \quad \mathbf{F} = \frac{\mathbf{Q}}{1 - \frac{\langle \mathbf{Q}^2 \rangle}{b}}, \tag{5.11}$$

for the Hookean, FENE and FENE-P models, respectively.

5.3.3 Polymer-contributed stress

- **Effective volume for stress averaging**

From the Lagrangian BDS point of view, the polymer contributed stress at each collocation point is determined by a local ensemble of dumbbells which are contained in a volume surrounding the collocation point, which is here termed ‘effective volume for stress averaging’ (EVSA), and the local velocity field. Therefore the convection of dumbbells in and out of the EVSA is tracked to determine the relevant ensemble of dumbbells. The knowledge of the local ensemble is a requirement for the numerical solution of the SDE to obtain the stress tensor. Fig. 5.1 shows schematically the EVSAs associated with collocation points. In this work, owing to the characteristic of the truly meshfree method, the EVSAs can be non overlapped or overlapped as shown in Fig. 5.1 to assure a sufficient number of dumbbells are used in the calculation of the polymer stress in the regions where the density of collocation points is very high. The size of the EVSA corresponding to a collocation point is dependent on the local density and the chosen number of dumbbells of each EVSA. The Voronoi tessellation technique could be exploited as a convenient way to build up EVSAs (Fig. 5.2) and this technique is supported by the library of several programming languages. Thus, Lagrangian particle tracking in FEM and in the present method are the same for either structured or unstructured discretisation, including the convection of dumbbells in and out of the EVSA (for the present method) or out of the element (for FEM). However, there are two principal differences between the two approaches. Firstly, there is no flexibility in determining the number of dumbbells in an element in the FEM whereas in the present method the number of dumbbells in the EVSAs is flexibly chosen, by virtue of the overlapping nature of EVSAs, to suit the local stress behaviour. Secondly, in the FEM, the average stress is considered constant in each element and is determined from the number of dumbbells located in the element whereas the stress is determined at collocation points in the present method. In the present work the minimum number of dumbbells in a EVSA is approximately 200 (with variance reduction).

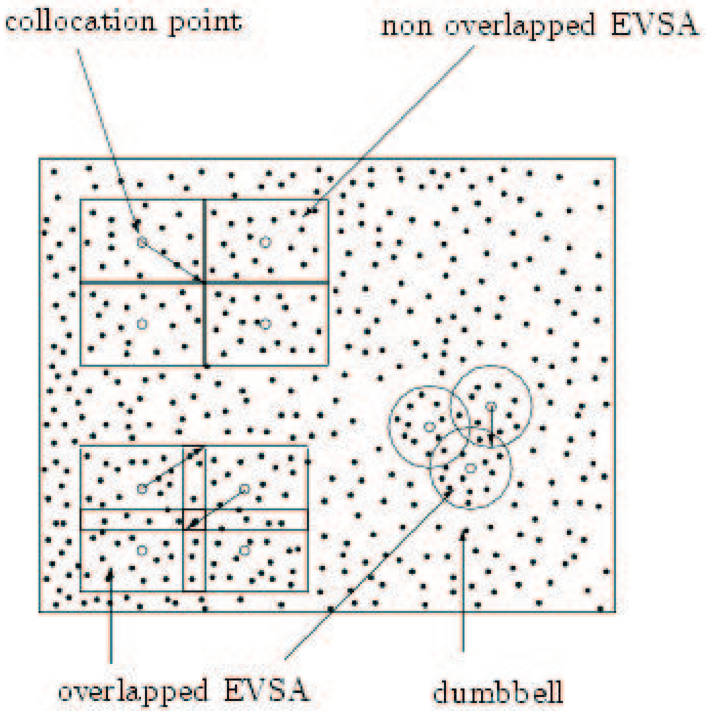


Figure 5.1: Effective volumes for stress averaging (EVSA): example of non-overlapped and overlapped EVSAs. 'o' denotes collocation points and '.' dumbbells. The collocation point and dumbbell distributions are only schematic.

- **Stress expression**

In order to evaluate the macroscopic stress tensor, an ensemble of N_d dumbbells is simulated. Once all $\mathbf{Q}s$ have been evaluated from the SDE (5.10), the relationship between the statistical distribution of dumbbell configurations and the polymer stress $\boldsymbol{\tau}^p$ at each collocation point $\mathbf{x}$ and time t are approximated by ensemble averages over the EVSAs described above and provided by Kramers' expression as follows

$$\boldsymbol{\tau}^p = -\left\langle \mathbf{Q} \cdot \mathbf{F}(\mathbf{Q}) \right\rangle + \mathbf{I}, \tag{5.12}$$

where $\mathbf{F}(\mathbf{Q})$ is dependent on the models and given by (5.11) for the Hookean, FENE and FENE-P models, respectively.

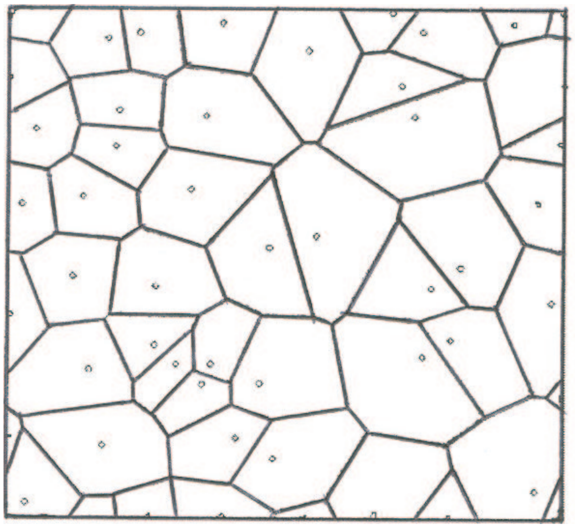


Figure 5.2: Effective volumes for stress averaging (EVSA): example of non-overlapped EVSAs using Voronoi Tessellation ‘o’ denotes collocation points. The collocation points can be random or regular.

5.4 Numerical method for SDEs

5.4.1 Numerical integration of SDEs

It is well known that Euler method does not produce accurate results in simulating 2-D visco-elastic fluid flow problems (Feigl *et al.*, 1995; Laso *et al.*, 1997; Halin 1998) and therefore Euler method is not employed in the present study of 2D problems. The numerical integration of the SDEs (5.7) or (5.10) is carried out by the second-order semi-implicit predictor-corrector scheme in this chapter. As described in §2.3.2, this scheme, as employed in many works on mesoscopic numerical simulation (Laso *et al.*, 1995, 1997; Halin *et al.*, 1998; Somasi and Khomami, 2000), consists of two steps, the predictor and corrector. For the FENE dumbbell model used as an illustration, from the Eqs (2.30) and (2.31), the two steps of the scheme are given by

- The predictor step. Let $\mathbf{Q}_i$, $\mathbf{Q}_{i+1}$ be an approximation of $\mathbf{Q}(t)$ at time t_i and t_{i+1}, respectively; $\Delta t_i = t_{i+1} - t_i = \textit{constant}$ is the stochastic time increment. The predictor step is given by

$$\mathbf{Q}^*_{i+1} = \mathbf{Q}_i + \left(\boldsymbol{\kappa}_i \cdot \mathbf{Q}_i - \frac{\mathbf{Q}_i}{2\left(1 - \frac{\mathbf{Q}_i^2}{b}\right)} \right) \Delta t_i + \Delta \mathbf{W}_i, \tag{5.13}$$

  where $\Delta \mathbf{W}_i = \mathbf{W}_{i+1} - \mathbf{W}_i$ whose components are independent Gaussian variables with zero mean and Δt variance (here the time step size is fixed $\Delta t_i = \Delta t$).

- The corrector step. This step is given by

$$\left(1 + \frac{\Delta t}{4\left(1 - \frac{\mathbf{Q}_{i+1}^2}{b}\right)}\right) \mathbf{Q}_{i+1} =$$
$$\mathbf{Q}_i + \frac{1}{2}\left(\boldsymbol{\kappa}^*_{i+1} \cdot \mathbf{Q}^*_{i+1} + \boldsymbol{\kappa}_i \cdot \mathbf{Q}_i - \frac{\mathbf{Q}_i}{2\left(1 - \frac{\mathbf{Q}_i^2}{b}\right)} \right) \Delta t_i + \Delta \mathbf{W}_i. \tag{5.14}$$

The direction of the corrected vector $\mathbf{Q}_{i+1}$ is determined by the direction of the RHS of (5.14), its length satisfies a cubic equation obtained from Eq (5.14) (see Appendix C.5); $\mathbf{Q}_{i+1}$ is unique and satisfies $0 \leq \|\mathbf{Q}_{i+1}\| < \sqrt{b}$ (Ottinger, 1996). The random numbers $\mathbf{W}_i$ in Eqs (5.13) and (5.14) are the same.

In this approach, the velocity gradient at t_{i+1} is required to be known at t_i (see the RHS of Eq (5.14)). In this chapter, the backward difference scheme is employed to approximate $\boldsymbol{\kappa}^*_{i+1}$ as follows

$$\boldsymbol{\kappa}^*_{i+1} = \boldsymbol{\kappa}_i + \dot{\boldsymbol{\kappa}}_i \Delta t_i \approx \boldsymbol{\kappa}_i + \frac{\boldsymbol{\kappa}_i - \boldsymbol{\kappa}_{i-1}}{\Delta t_{i-1}} \Delta t_i. \tag{5.15}$$

5.4.2 Integration of the trajectories of the dumbbells

Although the 2D flow is macroscopically a steady-state flow, the microscopic scheme requires the knowledge of the position of each dumbbell at every time step. An effective local volume is chosen to surround the collocation point, where the dumbbell is located, to calculate the macroscopic expectation stresses. For every dumbbell that is convected out of the volume under consideration at the outlet, a corresponding dumbbell is necessarily introduced at the inlet.
The centre of mass of each dumbbell moves along a trajectory that satisfies Eq (5.9) whose numerical solution is given according to the explicit Euler scheme as follows

$$x_{k,(i+1)} = x_{k,i} + u_k(\mathbf{x}, t_i)\Delta t_i, \qquad k = 1, 2, \tag{5.16}$$

where $x_{k,i}$ is the k^{th} component of the coordinate of the dumbbell at the time t_i; u_k is the k^{th} component of the velocity vector of the dumbbell. Since the dumbbells are convected by the bulk velocity field which is calculated at each time step, the trajectories of dumbbells can be determined. Details can be found in Luo and Mitsoulis (1990) and Laso *et al.* (1997).

5.4.3 Variance reduced simulation

As mentioned in chapter 2, in polymer dynamics, the variance reduced simulation achieves a reduction of the variance for the same outcome regarding the average of the quantities of interest without increasing the number of dumbbells (Melchior and Öttinger, 1995,1996). The variance reduction method consists of several different techniques which are detailed in §2.4.4, chapter 2. In this chapter, the control variate method is employed in conjunction with the Lagrangian CVFNNSS. This technique is considered as an unbiased estimator of $\langle . \rangle$ whose variance might be reduced via a good choice for the control variable. The method has been applied in other studies more recently (Jendrejack *et al.*, 2000; Kröger *et al.*, 2000 and Prabhakar and Prakash, 2002). Discussions on the efficiency of the scheme can be found in those references cited earlier, for example Melchior and Öttinger (1996), and are not repeated here.

For illustration, the scheme is described for the numerical approximation of the expectation of the random term $(\mathbf{Q} \cdot \mathbf{F}(\mathbf{Q}))$ in the following polymer stress formula (5.12)

$$\boldsymbol{\tau}^p = - \langle \mathbf{Q} \cdot \mathbf{F}(\mathbf{Q}) \rangle + \mathbf{I}, \tag{5.17}$$

where $\mathbf{Q}$ is governed by Eq (5.10)

$$d\mathbf{Q}(\mathbf{x}, t) = \left[\boldsymbol{\kappa}(\mathbf{x}, t) \cdot \mathbf{Q}(\mathbf{x}, t) - \frac{1}{2}\mathbf{F}(\mathbf{Q}(\mathbf{x}, t)) \right] dt + d\mathbf{W}(t). \tag{5.18}$$

The control variable is determined from a simulating process which is carried out in parallel with the process that approximates $\mathbf{Q}$.

At each position $\mathbf{x}$ and time $t > 0$, $\bar{\mathbf{Q}}$ corresponding to the connector vector $\mathbf{Q}$ is determined by

$$d\bar{\mathbf{Q}}(\mathbf{x}, t) = \left[\boldsymbol{\kappa}(\mathbf{x}, t) \cdot \bar{\mathbf{Q}}(\mathbf{x}, t) - \frac{1}{2}\bar{\mathbf{F}}(\bar{\mathbf{Q}}(\mathbf{x}, t)) \right] dt + d\mathbf{W}(t). \tag{5.19}$$

In this scheme, the expectation $\langle \mathbf{Q} \cdot \mathbf{F}(\mathbf{Q}) \rangle$ is split as follows

$$\langle \mathbf{Q} \cdot \mathbf{F}(\mathbf{Q}) \rangle = \langle \mathbf{Q} \cdot \mathbf{F}(\mathbf{Q}) - \bar{\mathbf{Q}} \cdot \bar{\mathbf{F}}(\bar{\mathbf{Q}}) \rangle + \langle \bar{\mathbf{Q}} \cdot \bar{\mathbf{F}}(\bar{\mathbf{Q}}) \rangle, \tag{5.20}$$

where $\bar{\mathbf{F}}(\bar{\mathbf{Q}})$ is determined by Eq (5.11). $\langle \bar{\mathbf{Q}} \cdot \bar{\mathbf{F}}(\bar{\mathbf{Q}}) \rangle$ is obtained from a deterministic process and $\langle \mathbf{Q} \cdot \mathbf{F}(\mathbf{Q}) - \bar{\mathbf{Q}} \cdot \bar{\mathbf{F}}(\bar{\mathbf{Q}}) \rangle$ by the stochastic simulation. A good choice of $\bar{\mathbf{Q}}$ makes reduce the variance corresponding to the RHS of Eq (5.20). From (5.20), the polymer stress tensor (5.17) is rewritten as follows

$$\boldsymbol{\tau}^p = -\langle \mathbf{Q} \cdot \mathbf{F}(\mathbf{Q}) - \bar{\mathbf{Q}} \cdot \bar{\mathbf{F}}(\bar{\mathbf{Q}}) \rangle + \bar{\boldsymbol{\tau}}^p, \tag{5.21}$$

where $\bar{\boldsymbol{\tau}}^p = -\langle \bar{\mathbf{Q}} \cdot \bar{\mathbf{F}}(\bar{\mathbf{Q}}) \rangle + \mathbf{I}$. At the equilibrium state, $\mathbf{u} = \mathbf{0}$ and $\bar{\mathbf{F}}(\bar{\mathbf{Q}}) = \mathbf{F}(\bar{\mathbf{Q}})$ (i.e. $\langle \bar{\mathbf{Q}} \cdot \mathbf{F}(\bar{\mathbf{Q}} \rangle = \mathbf{I}$, hence $\bar{\boldsymbol{\tau}}^p = \mathbf{0}$), the evolution equation of $\bar{\mathbf{Q}}$ is given by

$$d\bar{\mathbf{Q}}(\mathbf{x}, t) = -\frac{1}{2}\mathbf{F}\left(\bar{\mathbf{Q}}(\mathbf{x}, t)\right) dt + d\mathbf{W}(t). \tag{5.22}$$

Eq (5.22) can be solved by the Euler explicit scheme described in §2.3.2. Using the same random numbers for the realizations $\mathbf{Q}_i$ and $\bar{\mathbf{Q}}_i$ of $\mathbf{Q}$ and $\bar{\mathbf{Q}}$ ensures a strong correlation between them.

5.5 Review of RBFN interpolation

Unlike the RBFN used in the previous chapters, due to the complexity of the 2D problems, a combination of RBF and Polynomial Basis Function (PBF) is employed in a RBFN as follows (Fig. 5.3)

$$f(\mathbf{x}) = \sum_{j=1}^{m} w^j h^j(\mathbf{x}) + \sum_{k=1}^{\bar{m}} \iota^k p^k(\mathbf{x}) = \mathbf{H}^T(\mathbf{x})\mathbf{w} + \mathbf{P}^T(\mathbf{x})\boldsymbol{\iota}, \tag{5.23}$$

where $w^j \in \mathbf{w}$ ($\mathbf{w}^T = [w^1 \quad w^2 \quad ... \quad w^m]$) and $\iota^k \in \boldsymbol{\iota}$ ($\boldsymbol{\iota}^T = [\iota^1 \quad \iota^2 \quad ... \quad \iota^{\bar{m}}]$) are the synaptic weights; h^j is the chosen radial basis function corresponding to the j^{th} RBF-neuron; p^k is the polynomial basis function corresponding to the k^{th} PBF-neuron; $m + \bar{m}$ is the total number of neurons. $\mathbf{H}$ and $\mathbf{P}$ are defined as follows

$$\begin{aligned} \mathbf{H}^T(\mathbf{x}) &= [h^1(\mathbf{x}) \quad h^2(\mathbf{x}) \quad ... \quad h^m(\mathbf{x})], & (5.24) \\ \mathbf{P}^T(\mathbf{x}) &= [p^1(\mathbf{x}) \quad p^2(\mathbf{x}) \quad ... \quad p^{\bar{m}}(\mathbf{x})]. & (5.25) \end{aligned}$$

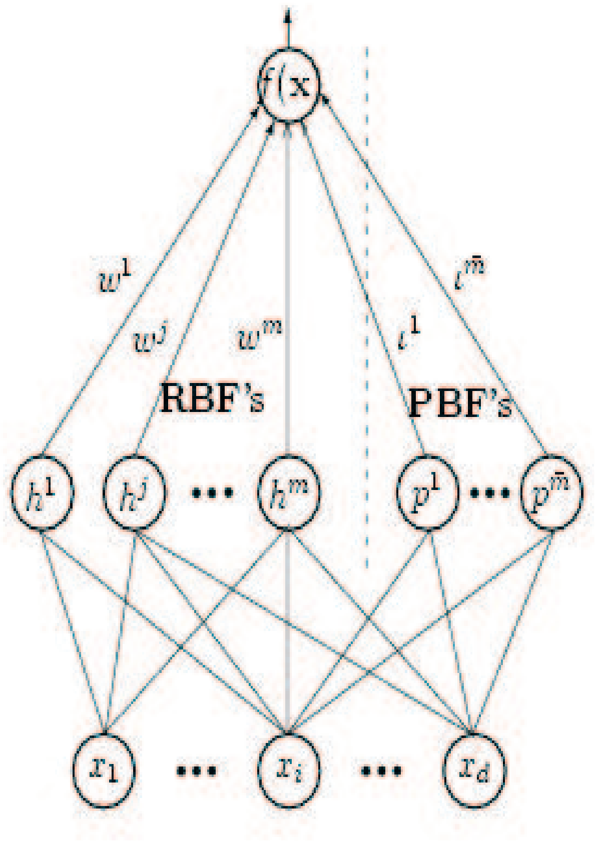


Figure 5.3: Schematic of the RBF-PBF neural networks. Each of d ($d = 2$, in the present work) components of the input vector $\mathbf{x}$ feeds forward to m RBFs and $\bar{m}$ PBFs whose outputs are linearly combined with weights $\{w^j\}_{j=1}^{m}$ and $\{\iota^k\}_{k=1}^{\bar{m}}$, respectively, into the network output $f(\mathbf{x})$.

Let n be the number of training points $(\mathbf{x}_i, \hat{y}_i)$; $\mathbf{x}_i$ is the coordinate of the i^{th} collocation point and $\hat{y}_i$ is the desired value of function f at the collocation point $\mathbf{x}_i$. While the RBFs h^j employed here are either MQ-RBF or TPS-RBF which were described in §4.5.1, the PBF of order p for a 2D problem is given by

$$\mathbf{P}^T(\mathbf{x}) = [1 \quad x_1 \quad x_2 \quad x_1^2 \quad x_1x_2 \quad x_2^2 \quad ... \quad x_2^p]. \tag{5.26}$$

5.5.1 RBF-centres, collocation points and RBFN training

The choice of the quantity and location of collocation points ($\mathbf{x}_i,\ i = 1,..,n$) depends on the problem geometry and desired solution accuracy and is a major open issue requiring separate investigation (Fodoseyev *et al*, 2002; Orr, 1999a,b; Larsson and Fornberg, 2001). However, one can imagine an analogy between an adaptive discretisation in the present finite point method and that in a finite element method. In this respect, an advantage of the present finite point method is that points can be added or removed much more easily than a corresponding addition or removal of finite elements, since there is no topology to be concerned about. In general, both collocation points and RBF centres can be randomly and separately distributed in the analysis domain. However, in the present work, collocation points are chosen to be the same as RBF centres, i.e. $m = n$ (Fig. 5.4). An extra requirement for the uniqueness of the approximation (5.23) is imposed as follows (Zerroukat *et al.*, 1998; Golberg *et al.*, 1996)

$$\sum_{i=1}^{n} p^k(\mathbf{x}_i)\omega_i = 0, \quad k = 1, \cdots, \bar{m}, \tag{5.27}$$

where the linear PBF is chosen ($\bar{m} = 3$), i.e

$$\mathbf{P}^T(\mathbf{x}) = [1 \quad x_1 \quad x_2]. \tag{5.28}$$

The training of the model Eq. (5.23), given a training set of n collocation points $\{(\mathbf{x}_i, \hat{y}_i)\}_{i=1}^{n}$, can be achieved via the minimisation of a cost function based on the sum of squared errors (taking into account (5.27))

$$C(\mathbf{w}^{'}, \lambda) = \sum_{i=1}^{n} (\hat{y}_i - f(\mathbf{x}_i))^2 + \lambda \sum_{j=1}^{n} w^{(j)2}, \tag{5.29}$$

with respect to the weights $\tilde{w} \in \mathbf{w'}, \mathbf{w'} = [\mathbf{w}^T, \boldsymbol{\iota}^T]^T$, which yields the following linear system of equations

$$\mathbf{G}\mathbf{w}^{'} = \mathbf{B}^T\hat{\mathbf{y}}^{'}, \tag{5.30}$$

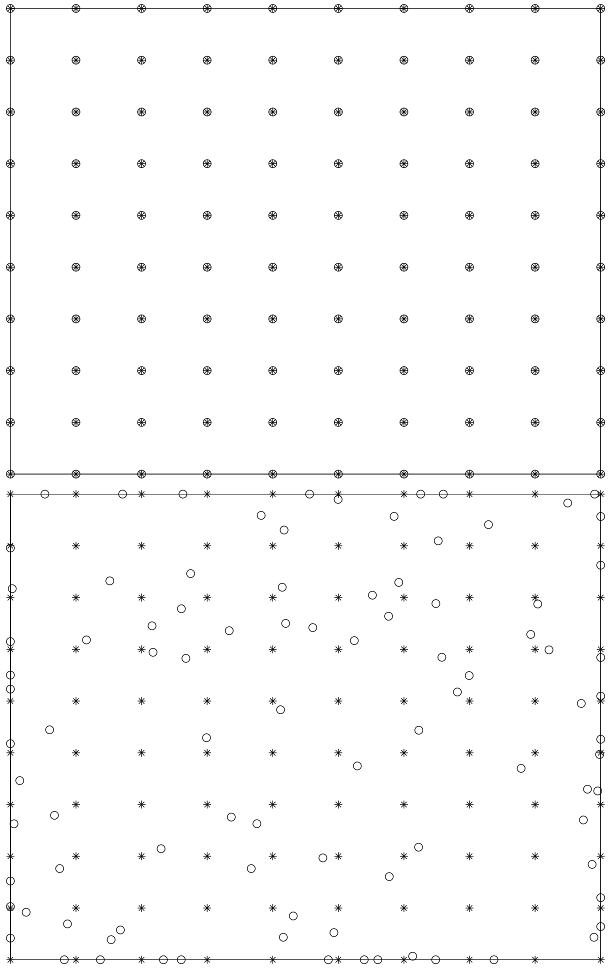


Figure 5.4: Collocation points 'o' and RBF centres '*': RBF centres are regularly distributed and collocation points are the same as RBF centres (upper figure); RBF centres are regularly distributed and collocation points are distributed randomly (lower figure).

where $\mathbf{G} = \mathbf{B}^T\mathbf{B} + \lambda\mathbf{K}$; λ is the global regularization parameter and $\mathbf{B}$ is given by

$$\mathbf{B} = \begin{bmatrix} \mathbf{H}^* & \mathbf{P}^* \\ \mathbf{P}^{*T} & \mathbf{0} \end{bmatrix},$$

$$\mathbf{H}^* = \begin{bmatrix} h^1(\mathbf{x}_1) & h^2(\mathbf{x}_1) & \cdots & h^n(\mathbf{x}_1) \\ h^1(\mathbf{x}_2) & h^2(\mathbf{x}_2) & \cdots & h^n(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ h^1(\mathbf{x}_n) & h^2(\mathbf{x}_n) & \cdots & h^n(\mathbf{x}_n) \end{bmatrix},$$

$$\mathbf{P}^* = \begin{bmatrix} p^1(x_1) & p^2(x_1) & p^3(x_1) \\ p^1(x_2) & p^2(x_2) & p^3(x_2) \\ \vdots & \vdots & \vdots \\ p^1(x_n) & p^2(x_n) & p^3(x_n) \end{bmatrix}, \quad \mathbf{K} = \begin{bmatrix} \mathbf{I}_n & \mathbf{0} \\ \mathbf{0} & \mathbf{0}_3 \end{bmatrix}, \tag{5.31}$$

$$\begin{aligned} \mathbf{w}^{'} &= [w^1 \quad w^2 \quad ... \quad w^n \quad \iota^1 \quad \iota^2 \quad \iota^3]^T \quad \text{and} \\ \hat{\mathbf{y}}^{'} &= [\hat{y}^1 \quad \hat{y}^2 \quad ... \quad \hat{y}^n \quad 0 \quad 0 \quad 0]^T. \end{aligned}$$

The partial derivatives of $f(\mathbf{x})$ can be calculated analytically as follows

$$\mathcal{L}f(\mathbf{x}) = \sum_{j=1}^{n} w^j \mathcal{L}h^j(\mathbf{x}) + \sum_{k=1}^{3} \iota^k \mathcal{L}p^k(\mathbf{x}), \tag{5.32}$$

where $\mathcal{L}$ is a derivative operator.

5.5.2 RBFN-based method for solving the deterministic PDEs

Considering the following problem in general

$$\mathcal{L}u = y \quad \forall x \in \Omega, \tag{5.33}$$

$$\mathcal{B}u = b \quad \forall x \in \Gamma, \tag{5.34}$$

where Ω is the volume under consideration; Γ is the boundary of Ω; $\mathcal{L}$ and $\mathcal{B}$ are differential operators; u is an unknown function; y and b are given functions. The problem may be multidimensional, however, without loss of generality, 1-D problem is presented here for simplicity. Since RBFN is a good universal approximator as mentioned above, u is represented by an approximant f defined by Eq (5.23) and then $\mathcal{L}f$ and $\mathcal{B}f$ can be described by (5.32). The numerical problem is therefore reduced to an unconstrained optimization problem of the objective function as follow

$$\Phi = \int_{\Omega} \parallel \mathcal{A}f - y \parallel^2 d\Omega + \int_{\Gamma_t} \parallel \mathcal{B}f - b \parallel^2 d\Gamma. \tag{5.35}$$

For a more detailed mathematical derivation leading to (5.35) see Franke (1982); Hornik *et al.* (1989); Golberg *et al.* (1996) and Dissanayake and Phan-Thien (1994). Equation (5.35) can now be applied particularly to solve the macroscopic governing equations.

5.5.3 RBFN-least-squares method for solving the continuity and momentum equations

In the continuity and momentum equations, the velocity field $\mathbf{u}$ and p are considered as unknowns while the polymer contributed stress $\boldsymbol{\tau}^p$ is already computed by the SST as described in §5.2. In this section, the numerical solution of Eqs (5.1) and (5.4) is obtained by a RBFN-based element free method.

In component forms, Eqs (5.1) and (5.2) in 2D space are as follows

$$\begin{aligned}
\frac{\partial u_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} &= 0, \\
\eta_N \left[\frac{\partial^2 u_1}{\partial x_1^2} + \frac{\partial^2 u_1}{\partial x_2^2}\right] - \rho \left[u_1 \frac{\partial u_1}{\partial x_1} + u_2 \frac{\partial u_1}{\partial x_2}\right] - \frac{\partial p}{\partial x_1} &= -\frac{\partial \tau_{11}^p}{\partial x_1} - \frac{\partial \tau_{21}^p}{\partial x_2}, \\
\eta_N \left[\frac{\partial^2 u_2}{\partial x_1^2} + \frac{\partial^2 u_2}{\partial x_2^2}\right] - \rho \left[u_1 \frac{\partial u_2}{\partial x_1} + u_2 \frac{\partial u_2}{\partial x_2}\right] - \frac{\partial p}{\partial x_2} &= -\frac{\partial \tau_{12}^p}{\partial x_1} - \frac{\partial \tau_{22}^p}{\partial x_2},
\end{aligned} \qquad (5.36)$$

and Eq (5.4) is developed as follows

$$\begin{aligned}
\eta_N \left[\frac{\partial^2 u_1}{\partial x_1^2} + \frac{\partial^2 u_1}{\partial x_2^2}\right] - \rho \left[u_1 \frac{\partial u_1}{\partial x_1} + u_2 \frac{\partial u_1}{\partial x_2}\right] + p_e \left[\frac{\partial^2 u_1}{\partial x_1^2} + \frac{\partial^2 u_1}{\partial x_1 \partial x_2}\right] &= \\
-\frac{\partial \tau_{11}^p}{\partial x_1} - \frac{\partial \tau_{21}^p}{\partial x_2}&, \\
\eta_N \left[\frac{\partial^2 u_2}{\partial x_1^2} + \frac{\partial^2 u_2}{\partial x_2^2}\right] - \rho \left[u_1 \frac{\partial u_2}{\partial x_1} + u_2 \frac{\partial u_2}{\partial x_2}\right] + p_e \left[\frac{\partial^2 u_2}{\partial x_2^2} + \frac{\partial^2 u_2}{\partial x_1 \partial x_2}\right] &= \\
-\frac{\partial \tau_{12}^p}{\partial x_1} - \frac{\partial \tau_{22}^p}{\partial x_2}&,
\end{aligned} \qquad (5.37)$$

together with the following boundary conditions

$$\begin{aligned}
u_1 &= u_o, \quad \mathbf{x} \in \Gamma_u, \\
\mathbf{n} \cdot \boldsymbol{\nabla} u_2 &= q_o, \quad \mathbf{x} \in \Gamma_t,
\end{aligned} \qquad (5.38)$$

where η_N is the viscosity of the solvent which is a Newtonian fluid; ρ is the density of the fluid; p is the pressure; $\mathbf{n}$ is the outward unit normal; u_o and q_o are known boundary conditions; $\Gamma = \Gamma_1 \cup \Gamma_2$. Since τ^p_{11}, τ^p_{12}, τ^p_{22} are the components of the polymer-contributed stress tensor which are considered as pseudo-forces. These are calculated from stochastic simulation technique, the terms on the RHS of Eqs (5.36) or (5.37) are known and calculated as follows

(1.) τ^p_{11}, τ^p_{12}, τ^p_{22}, calculated by the SST, are approximated by three separate RBFNs;

(2.) The derivatives of the polymer stresses in Eqs. (5.36) or (5.37) are then calculated analytically as shown in §5.5.

Equations (5.36) or (5.37) and (5.38) can now be discretised by a process of point collocation. As mentioned in §5.5.1, generally, the collocation points in the domain and on the boundary can be different from the RBF centres. The centres can be random or arranged on a regular grid. The set of collocation points can be different from, the same as, or a subset of the set of RBF centres. In this work, the two sets are the same and hence $n = m$ where n is the number of collocation points and m is the number of neurons of RBFNs.

With a chosen set of collocation points (in the analysis volume and on the boundaries), according to Eq (5.35), the sum square error associated with Eqs (5.36) and (5.38) is written as follows

$$\begin{aligned}
SSE = & \sum_{\mathbf{i}\in\Omega}\left\{\frac{\partial u_1}{\partial x_1}+\frac{\partial u_2}{\partial x_2}\right\}^2_{\mathbf{i}} \\
& + \sum_{\mathbf{i}\in\Omega}\left\{\eta_N\left[\frac{\partial^2 u_1}{\partial x_1^2}+\frac{\partial^2 u_1}{\partial x_2^2}\right]-\rho\left[u_1\frac{\partial u_1}{\partial x_1}+u_2\frac{\partial u_1}{\partial x_2}\right]-\frac{\partial p}{\partial x_1}+\left[\frac{\partial \tau^p_{11}}{\partial x_1}+\frac{\partial \tau^p_{21}}{\partial x_2}\right]\right\}^2_{\mathbf{i}} \\
& + \sum_{\mathbf{i}\in\Omega}\left\{\eta_N\left[\frac{\partial^2 u_2}{\partial x_1^2}+\frac{\partial^2 u_2}{\partial x_2^2}\right]-\rho\left[u_1\frac{\partial u_2}{\partial x_1}+u_2\frac{\partial u_2}{\partial x_2}\right]-\frac{\partial p}{\partial x_2}+\left[\frac{\partial \tau^p_{12}}{\partial x_1}+\frac{\partial \tau^p_{22}}{\partial x_2}\right]\right\}^2_{\mathbf{i}} \\
& + \sum_{\mathbf{i}\in\Gamma_u}\{u_1-u_o\}^2+\sum_{\mathbf{i}\in\Gamma_t}\left\{[n_1\frac{\partial u_1}{\partial x_1}+n_2\frac{\partial u_1}{\partial x_2}]-q_o\right\}^2_{\mathbf{i}},
\end{aligned} \quad (5.39)$$

and the sum square error associated with Eqs (5.37) and (5.38) is given by

$$
\begin{aligned}
&SSE = \\
&\sum_{\mathbf{i}\in\Omega}\left\{\eta_N\left[\frac{\partial^2 u_1}{\partial x_1^2}+\frac{\partial^2 u_1}{\partial x_2^2}\right]-\rho\left[u_1\frac{\partial u_1}{\partial x_1}+u_2\frac{\partial u_1}{\partial x_2}\right]+p_e\left[\frac{\partial^2 u_1}{\partial x_1^2}+\frac{\partial^2 u_1}{\partial x_1\partial x_2}\right]_i+\phi_{1i}\right\}_{\mathbf{i}}^2+\\
&\sum_{\mathbf{i}\in\Omega}\left\{\eta_N\left[\frac{\partial^2 u_2}{\partial x_1^2}+\frac{\partial^2 u_2}{\partial x_2^2}\right]-\rho\left[u_1\frac{\partial u_2}{\partial x_1}+u_2\frac{\partial u_2}{\partial x_2}\right]+p_e\left[\frac{\partial^2 u_2}{\partial x_2^2}+\frac{\partial^2 u_2}{\partial x_1\partial x_2}\right]_i+\phi_{2i}\right\}_{\mathbf{i}}^2+\\
&\qquad\qquad\sum_{\mathbf{i}\in\Gamma_u}\{u_1-u_o\}^2+\sum_{\mathbf{i}\in\Gamma_t}\left\{[n_1\frac{\partial u_1}{\partial x_1}+n_2\frac{\partial u_1}{\partial x_2}]-q_o\right\}_{\mathbf{i}}^2, \qquad (5.40)
\end{aligned}
$$

where i denotes the i^{th} collocation point; $\phi_{1i} = \frac{\partial\tau_{11}^p}{\partial x_1}(\mathbf{x}_i)+\frac{\partial\tau_{21}^p}{\partial x_2}(\mathbf{x}_i)$ and $\phi_{2i} = \frac{\partial\tau_{12}^p}{\partial x_1}(\mathbf{x}_i)+\frac{\partial\tau_{22}^p}{\partial x_2}(\mathbf{x}_i)$. Substituting the expression of u_1, u_2, p (without penalty method) and their derivatives, (given by Eqs (5.23), (5.32), (3.35), (4.19)) into Eq (5.39) and applying the general linear least squares principle, a system of linear algebraic equations of the unknown weights is obtained (see §5.5). Although the solution can be obtained by several schemes mentioned in previous chapters, here the algorithm to solve the least squares problem (5.39) or (5.40) is the Singular Value Decomposition (SVD) which can be found in Trefethen and Bau Iii (1997) and Bjock (1996) (see appendix B.4).

The non-linear convective term $(\mathbf{u}\cdot\nabla)\mathbf{u}$ present in (5.39) or (5.40) is estimated using a Picard-type iterative procedure as follows

(1.) Imposing the initial velocity field for the first iteration (zero value in this work);

(2.) Linearizing the non-linear terms by using the current estimate of the velocity field, keeping the derivatives as unknown, i.e. $(\mathbf{u}\cdot\nabla)\mathbf{u}$ is represented by $(\mathbf{u}^n\cdot\nabla)\mathbf{u}^{n+1}$;

(3.) Obtaining a new estimate of the velocity field via (5.39) or (5.40) in the sense of the general linear least squares principle;

(4.) Calculating the convergence measure (CM) for velocity at the n^{th} iteration, defined as follows

$$CM = \sqrt{\frac{\sum_{j=1}^{N}\sum_{i=1}^{2}(u_{i,j}^{n} - u_{i}^{n-1})^2}{\sum_{j=1}^{N}\sum_{i=1}^{2}(u_{i,j}^{n})^2}} < tol, \tag{5.41}$$

where N is the number of collocation points;

(5.) If not yet converged ($CM > tol$; tol is a tolerance), return to step 2;

(6.) Stop.

5.6 Algorithm of the present method

In summary, the present approach can be described in a detailed algorithm as follows.

(1.) Generate a set of collocation points. Start with an initial velocity for the first iteration (zero in the present work) and the boundary conditions of the problem. The initial velocity field is approximated by RBFNs;

(2.) Generate an ensemble of homogenously distributed dumbbells in the flow domain. This initialisation of the polymer configuration field is based on the known equilibrium distribution function which is a three dimensional Gaussian distribution with zero mean and unit covariance (Öttinger, 1996). Determine an ensemble of control variates $\mathbf{Q}_c$'s associated with $\mathbf{Q}$'s as represented in §5.4.3;

(3.) Generate effective volumes for stress averaging (EVSA) surrounding the collocation points, i.e. only dumbbells belonging to a EVSA have any influence on the polymer contributed stress at the collocation point associated with that EVSA. Determine the EVSA where a dumbbell is initially located. In this work, each dumbbell can belong to one (non-overlapped case) or many different EVSA's (overlapped case) (Fig. 5.2);

(4.) After the current velocity field is approximated by RBF networks, determine the velocity gradient field by calculating directly the gradient of the approximated velocity field;

(5.) Calculate the polymer configuration field (the connector vectors of the dumbbell ensemble) using the methods described in §5.3 and the corresponding control variates as described in §5.4.3. The velocity of the centre of mass of each dumbbell is considered to be equal to the pointwise local fluid velocity;

(6.) If there are dumbbells leaving the domain under consideration at the outlet, they are relocated at the inlet with the same value of connector vector;

(7.) Determine the local stress tensor by taking the ensemble average of the polymer configuration in each effective volume using variance reduced simulation. Assign this stress to the collocation point associated with this volume and then impose the stress boundary conditions at the collocation points located on boundary, if relevant. The stress is then approximated globally by RBF networks which are the ultimate description of the stress field. This global approximation procedure smooths the piecewise continuous stress field with a globally continuous function. This could be achieved by either TPS-RBFNs or MQ-RBFNs. However, the former was proved to have superior smoothing characteristics (Beatson and Light, 1997) and hence is used in the present work;

(8.) With the stress field just obtained, solve the set of conservation equations for the new velocity field using the element-free RBFN method as described in §5.5.2; Determine the new positions of dumbbells and the EVSA where they are located (particle tracking).

(9.) Terminate the simulation when either the desired time or the convergence is reached. The latter is determined by a convergence measure for either the velocity field or the stress field between two consecutive iterations which is defined for the velocity field by

$$CM = \sqrt{\frac{\sum_1^N \sum_{i=1}^d (u_i^n - u_i^{n-1})^2}{\sum_1^N \sum_{i=1}^d (u_i^n)^2}} < tol, \tag{5.42}$$

where d is the number of dimensions (2 in the present work); tol is a preset tolerance; u_i is the i component of the velocity at a node; N is the total number of collocation points and n is the iteration number. Convergence is also checked for the shear stress and the first normal stress difference;

(10.) Return to step (4) for the next time level of the microscopic process.

5.7 Numerical examples

This section reports the verification of the present method with the simulation of the planar Poiseuille flow, and the application of the method to solve the benchmark 4:1 axisymmetric contraction flow problem. Kinetic dumbbell models considered include the Hookean, FENE and FENE-P models. These models consist of non-interacting dumbbells neutrally suspended in a Newtonian solvent of known viscosity η_N, density ρ. The resultant material is characterized by the relaxation time λ_H and zero-shear-rate viscosity $\eta_o = \eta_N + \eta_p$.

5.7.1 Planar Poiseuille flow

The planar creeping Poiseuille problem and coordinate system is described in Fig. 5.5a where only half of the fluid domain needs to be considered, owing to symmetry. For this problem, a is a half of the gap between the two parallel plates. The length of the domain under consideration is a. The domain under consideration is filled with 240000 dumbbells. In this example, the results of the creeping Poiseuille flow problem using Hookean dumbbell model are compared with the analytical result of the Oldroyd-B model. Furthermore, the results

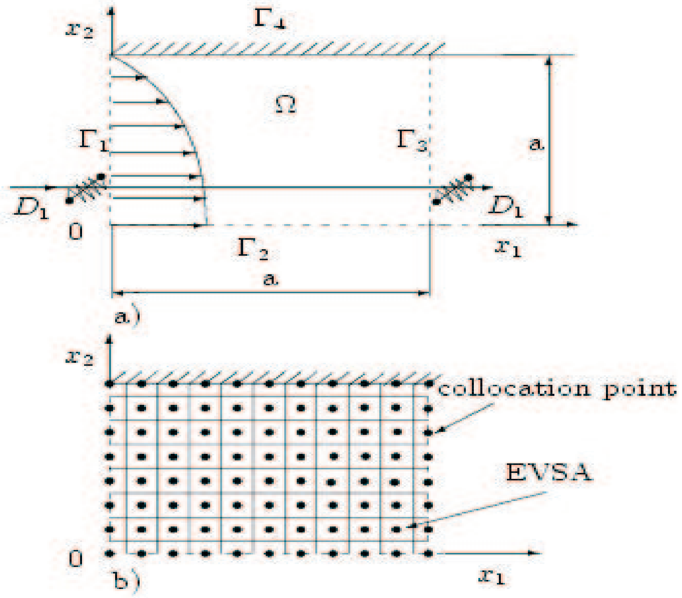


Figure 5.5: The planar Poiseuille flow problem: the inlet velocity profile is fully developed; non-slip boundary conditions applied at the fluid-solid interfaces. The collocation point distribution is only schematic.

of the FENE and FENE-P models indicate partially the behavior differences between these fluids. The fluid parameters are as follows (Feigl *et al.*, 1995)

$$\begin{aligned} \eta_N &= 0.5; \quad \alpha = \frac{\eta_N}{\eta_o} = 0.5; \quad \lambda_H = 1; \quad V = 1; \\ \text{and} \quad b &= 50 \quad \text{for both FENE and FENE-P models.} \end{aligned}$$

where V is the maximum velocity on the centreline corresponding to a given flow rate.

Boundary conditions and analytical solution

The macroscopic boundary conditions are given as follows:

1. On the wall (Γ_4), there is no slip

$$\mathbf{u}(\mathbf{x}) = 0;$$

2. At the inlet section (Γ_1), the flow is fully developed Poiseuille where the velocity profile is parabolic for the Newtonian or Boger fluid as follows

$$\begin{cases} u_1(\mathbf{x}) = u_p = V\left(1 - \left(\frac{x_2}{a}\right)^2\right), \\ u_2(\mathbf{x}) = 0; \end{cases}$$

   If the model is shear-thinning (for example the FENE model), this velocity profile is not parabolic and determined by the periodical boundary condition at inlet and outlet. Generally, the planar Poiseuille flow can be computed as a 1D-problem. In this work, the 2-D method described in Laso *et al.*, (1999) is employed as follows

   - Initially, the inlet of the domain is given a guessed initial velocity profile (zero-value or a parabolic profile as described above for Newtonian fluid in the present work);
   - The obtained outlet velocity profile at a step i is used to update the inlet velocity profile of the next step $i+1$.
   - The process is continued until there is no further change in the outlet profile.

3. At the outlet section (Γ_3)

$$u_2(\mathbf{x}) = 0,$$

4. On the centreline (Γ_2), the symmetry condition applies

$$u_2(\mathbf{x}) = 0, \quad \frac{\partial u_1}{\partial x_2}(\mathbf{x}) = 0, \quad \tau_{12} = 0.$$

Furthermore, in the microscopic component, the domain is filled with dumbbells drawn from the equilibrium distribution function. The dumbbells are convected according to the velocity field of flows. Dumbbells leaving the domain through the outlet are reintroduced at the inlet at the same value x_2 and the same configuration. Although the fully developed velocity of Poiseuille flows using Hookean (Oldroyld-B) dumbbell model can be approximated as parabolic profile, the scheme described above is used in the $4:1$ axisymmetric contraction flow problem described next in §5.7.2.

For the Hookean dumbbell (Oldroyd-B) model, the creeping Poiseuille flow problem has the analytical solution given by

$$\tau_{11} = 3(1-\alpha)De\left(\frac{\partial u_1}{\partial x_2}\right)^2; \quad \tau_{12} = (1-\alpha)\frac{\partial u_1}{\partial x_2}; \quad \tau_{22} = 0. \tag{5.43}$$

where $De = \lambda_H \frac{\langle u_1 \rangle}{a} = \frac{2}{3}\lambda_H \frac{V}{a} = \frac{2}{3}$ is the Deborah number. The above analytical solution is used to judge the quality of the numerical results.

Sum square error

The expression of sum square error (5.39) for the creeping planar Poiseuille flows is rewritten as follows

$$\begin{aligned}
SSE = \sum_{\mathbf{x}_i \in \Omega} &\left(\frac{\partial u_1}{\partial x_1}(\mathbf{x}_i) + \frac{\partial u_2}{\partial x_2}(\mathbf{x}_i)\right)^2 + \\
&\sum_{\mathbf{x}_i \in \Omega}\left(\eta_N\left[\frac{\partial^2 u_1}{\partial x_1^2}(\mathbf{x}_i) + \frac{\partial^2 u_1}{\partial x_2^2}(\mathbf{x}_i)\right] - \frac{\partial p}{\partial x_1}(\mathbf{x}_i) + \phi_1(\mathbf{x}_i)\right)^2 + \\
&\sum_{\mathbf{x}_i \in \Omega}\left(\eta_N\left[\frac{\partial^2 u_2}{\partial x_1^2}(\mathbf{x}_i) + \frac{\partial^2 u_2}{\partial x_2^2}(\mathbf{x}_i)\right] - \frac{\partial p}{\partial x_2}(\mathbf{x}_i) + \phi_2(\mathbf{x}_i)\right)^2 + \\
\sum_{\mathbf{x}_i \in \Gamma_1}\{u_1(\mathbf{x}_i) - u_p\}^2 + &\sum_{\mathbf{x}_i \in \Gamma_1} u_2^2(\mathbf{x}_i) + \sum_{\mathbf{x}_i \in \Gamma_2} u_2^2(\mathbf{x}_i) + \sum_{\mathbf{x}_i \in \Gamma_2}\left\{\frac{\partial u_1}{\partial x_2}(\mathbf{x}_i)\right\}^2 + \\
&\sum_{\mathbf{x}_i \in \Gamma_3} u_2^2(\mathbf{x}_i) + \sum_{\mathbf{x}_i \in \Gamma_4} u_1^2(\mathbf{x}_i) + \sum_{\mathbf{x}_i \in \Gamma_4} u_2^2(\mathbf{x}_i)
\end{aligned} \tag{5.44}$$

where $\phi_1(\mathbf{x}_i) = \frac{\partial \tau_{11}^p}{\partial x_1}(\mathbf{x}_i) + \frac{\partial \tau_{21}^p}{\partial x_2}(\mathbf{x}_i)$; $\phi_2(\mathbf{x}_i) = \frac{\partial \tau_{12}^p}{\partial x_1}(\mathbf{x}_i) + \frac{\partial \tau_{22}^p}{\partial x_2}(\mathbf{x}_i)$; u_p is the inlet velocity profile given above.

Results and discussion

Initially, an ensemble of dumbbells (240000) is homogenously distributed in the whole domain. The schematic distribution of collocation points and associated EVSA's are shown in Fig. 5.5b. In this example, the non-overlapped EVSA's are employed, using four sets of coarse density collocation points, namely (11×11),

Table 5.1: Planar Poiseuille flow: Collocation point convergence. N is the total number of collocation points; CR: the convergence of measure. The number of test points $N_t = 121$.

N	CR
11×11	1.00000
15×15	0.00104
19×19	0.00087
21×21	0.00082

(15×15), (19×19) and (21×21). The convergence test (which is described in §3.4.3 of chapter 3), using 121 test points, shows that the problem can be solved with a coarse grid of collocation points (Table 5.1). The Deborah number is chosen to be 2/3. Higher values of Deborah number can easily be simulated, but the results are of no further interest, and hence not reported here. The velocity and stress fields corresponding to a grid of collocation points 15×15, are obtained by averaging the results of 1000 iterations after the convergence measure CM_u reaches $1.0e-4$ for the velocity field. For the Hookean dumbbell model, Fig. 5.6 shows the velocity profile on the plane $x_1 = 0.5$ which is in excellent agreement with the inlet velocity profile. Fig. 5.7 shows the shear stress and the first normal stress difference on the plane $x_1 = 0.5$. Compared with the analytical results from Eq (5.43), the error of the shear stress is insignificant (see Fig. 5.7), and the error of the first normal stress difference is less than (2%). At the steady state, the velocity profiles and shear stresses of the two models (the FENE and FENE-P models) are nearly the same while the first normal stress differences are different (see Fig. 5.8). Similar conclusion was found, for example, in Laso and Öttinger (1993) and Tran-Canh and Tran-Cong (2002b) for the start-up planar Couette flow problem.

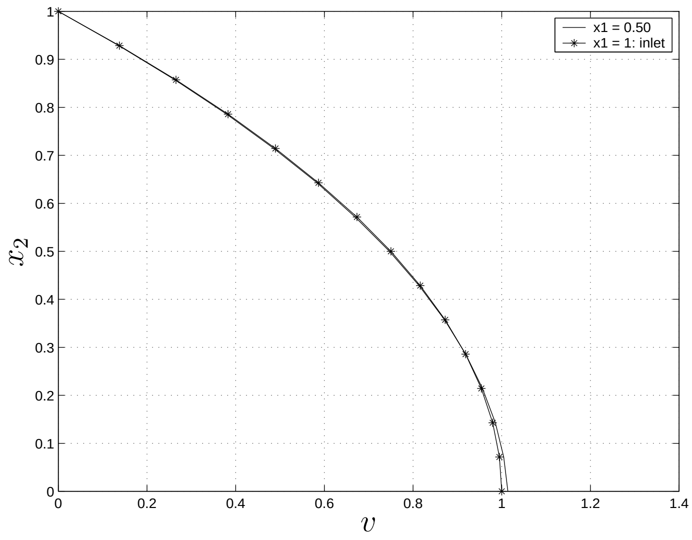


Figure 5.6: Steady-state planar Poiseuille using the Hookean dumbbell model: the velocity profile on the cross-section $x_1 = 0.5$ and at the inlet section with 15×15 collocation points.

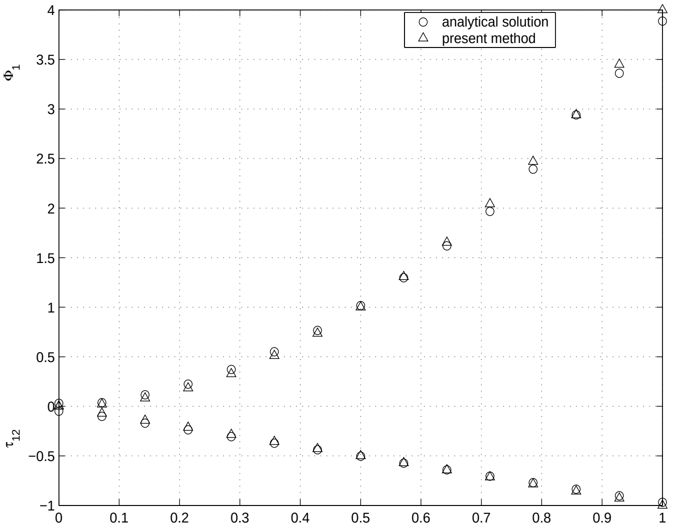


Figure 5.7: Steady-state planar Poiseuille flow problem using the Hookean dumbbell model: comparison of the polymer shear stress and the first normal stress difference obtained by the present method with the analytical solution on the plane $x_1 = 0.5$ with 15×15 collocation points.

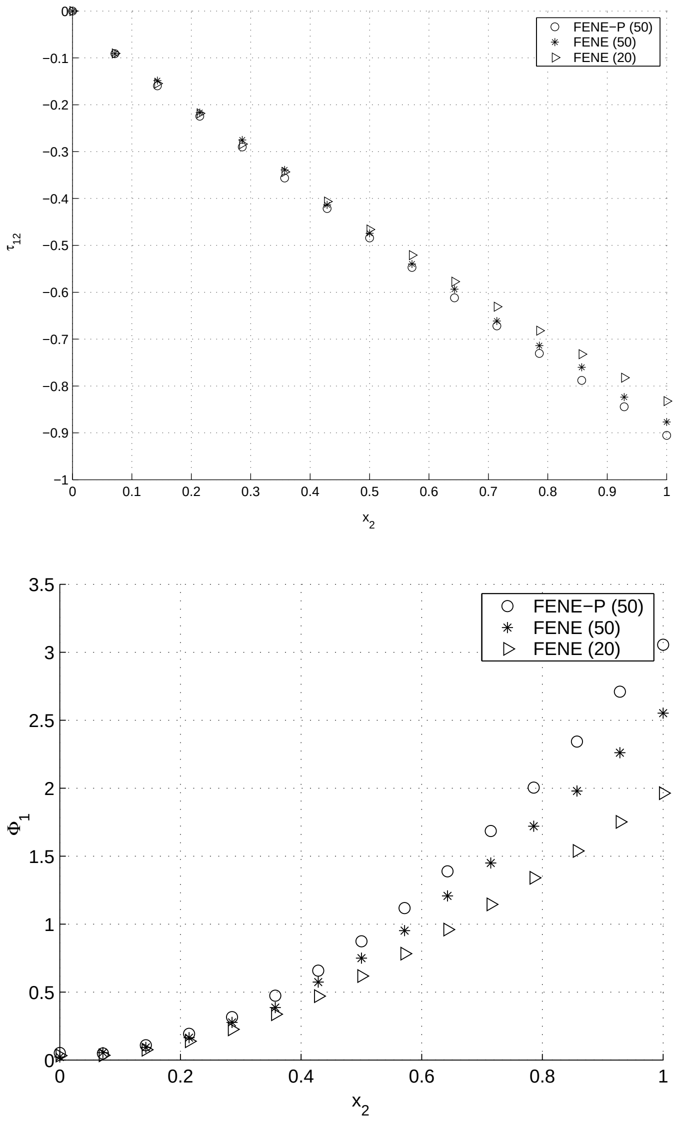


Figure 5.8: Steady-state planar Poiseuille flow problem using the FENE (b=50, b=20) and FENE-P (b=50) models: the polymer shear stress and the first normal stress difference on the cross-section $x_1 = 0.5$.

5.7.2 The 4:1 axisymmetric contraction flow

In this section, the ability of the present method is investigated using a benchmark problem: the steady state 4:1 axisymmetric contraction flow.

This steady state problem was simulated with different models and schemes (Marchal and Crochet, 1987; Debbaut *et al.*, 1988; Dupont and Crochet, 1988; Luo and Mitsoulis, 1990, Boger *et al.*, 1992 and Debae *et al.*, 1994). Especially Feigl *et al.* (1995) used the CONNFFESSIT method to solve this problem using Hookean model and compared their solution with the results obtained using the FEM (Bernstein *et al.*, 1994). For the sake of comparison with the results obtained from a hybrid simulation scheme (CONNFFESSIT), the same model and physical parameters as in Feigl *et al.* (1995) are used for this example.

Owing to symmetry, only half of the analysis domain needs to be considered in cylindrical coordinates as shown in Fig. 5.9. Let $r_d = 1$ be the radius of downstream channel. The radius of upstream channel is $r_u = 4r_d$; the upstream and downstream lengths are $L_u = 10r_d$ and $L_d = 30r_d$, respectively. The fluid parameters are given by

$$\rho = 0.1; \quad \eta_o = \eta_N + \eta_p = 1; \alpha = \frac{\eta_N}{\eta_p} = 1; \quad \lambda_H = 1, V_u = 0.3 \tag{5.45}$$

where V_u is the maximum inlet velocity on the centreline and η_o is the viscosity of the fluid. The boundary conditions of the problem are the ones which were employed in Dupont *et al.* (1988); Luo and Mitsoulis, (1990); Ryssel and Brunn (1999); Boger *et al.* (1992); Feigl *et al.* (1995) where no slip condition is imposed along the wall and a fully developed velocity profile is specified at the inlet and outlet. The symmetry condition is imposed on the centre-line. Let Γ be the boundary of the domain Ω; $\Gamma = \Gamma_1 \cup \Gamma_2 \cup \Gamma_3 \cup \Gamma_4$ as shown in Fig. 5.9. The boundary conditions are therefore given by

1. On the wall (Γ_4), there is no slip condition, i.e. $u_r = u_x = 0$

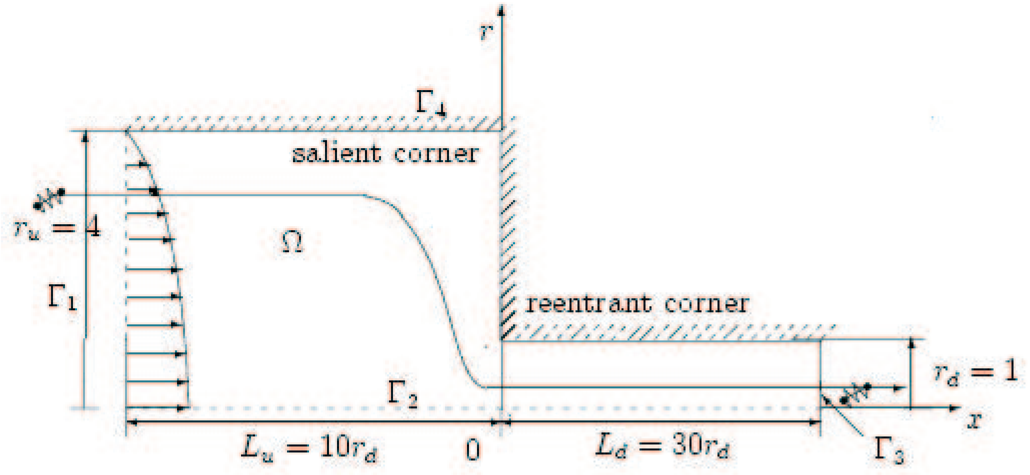


Figure 5.9: The steady state axisymmetric 4:1 contraction flow problem: non-slip boundary conditions are imposed at the fluid-solid interfaces; fully developed Poiseuille at the inlet, and symmetry on the centre-line.

2. At the inlet (Γ_1), the flow is fully developed. In this work, the velocity profile at the inlet is determined by solving the 2D Poiseuille problem (see §5.7.1) with the same V_u at the inlet centreline. The approximated velocity profile, and the dumbbells leaving the domain after convergence for a time period Δt are stored and used as the inlet condition in the 4:1 contraction flow problem. The obtained inlet velocity profile given in Fig. 5.10 shows that the difference between the velocity profiles of the parabolic approximation and present work is very little. At the outlet, $u_r = 0$ and u_x is considered as an unknown and need to be calculated.

3. On the centreline (Γ_2), the symmetry condition is given by

$$\frac{\partial u_x}{\partial r}(\mathbf{x}) = 0, \qquad u_r = 0. \tag{5.46}$$

Three sets of collocation points are employed for this problem and given in Table 5.2. The collocation points are not uniformly distributed and a higher density of collocation points is specified in the reentrant corner area to capture the expected strong flow gradients in that region. A schematic distribution of

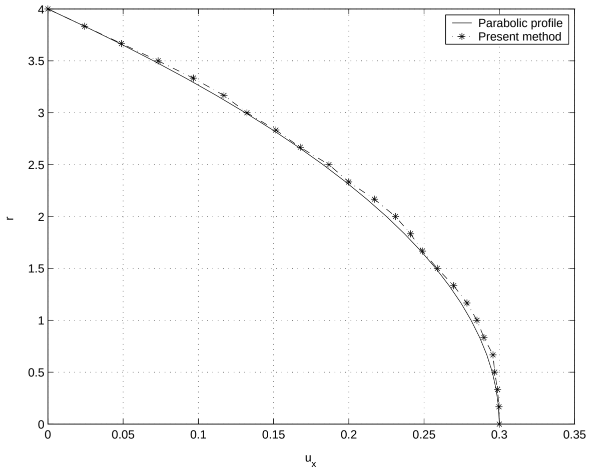


Figure 5.10: The axisymmetric 4:1 contraction problem: the inlet velocity profile using. The solid line denotes the parabolic approximation and '-*-' the results by the present method.

Table 5.2: The axisymmetric 4:1 contraction flow problem: three sets of collocation points. N_t: the total number of collocation points; N_{up}: the number of collocation points of the upstream area; N_{dn}: the number of collocation points of the downstream area; N_b: the number of boundary points.

N_t	N_{up}	N_{dn}	N_b
404	224	180	115
620	340	280	151
755	440	315	169

collocation points and associated EVSAs are shown in Fig. 5.11, using 450000 dumbbells in the total domain under consideration. In order to assure a sufficient number of dumbbells in each EVSA in the areas where the density of collocation points is high, the overlapped EVSA method is employed in the salient corner area and specially the reentrant corner area as shown in Fig. 5.11.

The penalty method leading to the form of momentum equation (5.6) is employed to reduce the number of degrees of freedom. Thus with regard to the

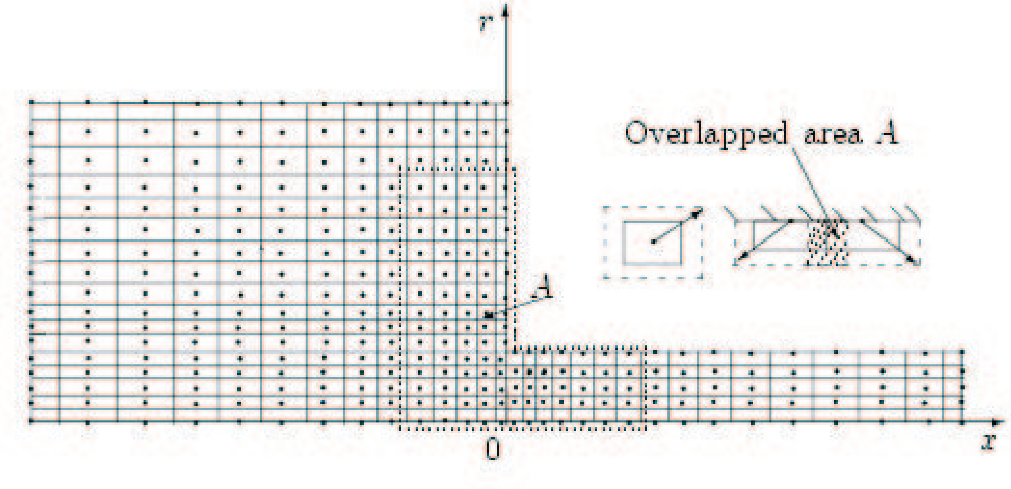


Figure 5.11: The steady state axisymmetric 4:1 contraction flow problem: schematic distribution of collocation points ‘ $\centerdot$’ and EVSAs.

symmetry, the sum square error (5.40) is given in the cylindrical coordinates by

$$\begin{aligned} SSE = & \sum_{i\in\Omega}\left\{\frac{\partial u_r}{\partial r}+\frac{u_r}{r}+\frac{\partial u_x}{\partial x}\right\}_i^2+\sum_{i\in\Omega}\left\{\eta_N\left[\frac{\partial^2 u_r}{\partial r^2}+\frac{1}{r}\frac{\partial u_r}{\partial r}+\frac{\partial^2 u_r}{\partial x^2}-\frac{u_r}{r^2}\right]_i\right. \\ & \left.-\rho\left[u_r\frac{\partial u_r}{\partial r}+u_x\frac{\partial u_r}{\partial x}\right]_i+p_e\left[\frac{\partial^2 u_r}{\partial r^2}+\frac{1}{r}\left(\frac{\partial u_r}{\partial r}-\frac{u_r}{r}\right)+\frac{\partial^2 u_r}{\partial x^2}\right]+\phi_{1i}\right\}_i^2 \\ & +\sum_{i\in\Omega}\left\{\eta_N\left[\frac{\partial^2 u_x}{\partial r^2}+\frac{1}{r}\frac{\partial u_x}{\partial r}+\frac{\partial^2 u_x}{\partial x^2}\right]_i-\rho\left[u_r\frac{\partial u_x}{\partial r}+u_x\frac{\partial u_x}{\partial x}\right]+p_e\left[\frac{1}{r}\frac{\partial u_x}{\partial r}\right.\right. \\ & \left.\left.+\frac{\partial^2 u_x}{\partial r^2}+\frac{\partial^2 u_x}{\partial x^2}\right]_i+\phi_{2i}\right\}^2+\sum_{i\in\Gamma_1}\{u_x-u_p\}_i^2+\sum_{i\in\Gamma_1}\{u_r\}_i^2+\sum_{i\in\Gamma_2}\{u_r\}_i^2 \\ & +\sum_{i\in\Gamma_2}\left\{\frac{\partial u_x}{\partial r}\right\}_i^2+\sum_{i\in\Gamma_3}\{u_r\}_i^2+\sum_{i\in\Gamma_4}\{u_r\}_i^2+\sum_{i\in\Gamma_4}\{u_x\}_i^2 \qquad (5.47) \end{aligned}$$

where $\phi_{1i}=\frac{\tau^p_{rr}-\tau^p_{\theta\theta}}{r}+\frac{\partial\tau^p_{rr}}{\partial r}+\frac{\partial\tau^p_{rx}}{\partial x}|_i$; $\phi_{2i}=\frac{\tau^p_{rx}}{r}+\frac{\partial\tau^p_{rx}}{\partial r}+\frac{\partial\tau^p_{xx}}{\partial x}|_i$; u_p is the inlet velocity profile given in the section on boundary conditions. The stochastic simulation is time-dependent with simulation time step $\Delta t = 0.01\lambda = 0.01$.

For large problems, the coefficient matrix, resulting from the application of the general linear least squares principle to the SSE, could be ill-conditioned. Such a coefficient matrix is also dense. To overcome the problem of ill-condition and

Table 5.3: Axisymmetric 4:1 contraction flow problem: two sets of collocation points. N: the number of subdomains; N_t: the number of collocation points; N_s: the number of collocation point of each subdomain and CM_u: convergence measure based on velocity.

N	N_t	N_s	CM_u
4	620	$145 - 170$	$1.52e - 4$
8	620	$77 - 85$	$2.12e - 4$
12	620	$50 - 60$	$2.68e - 4$

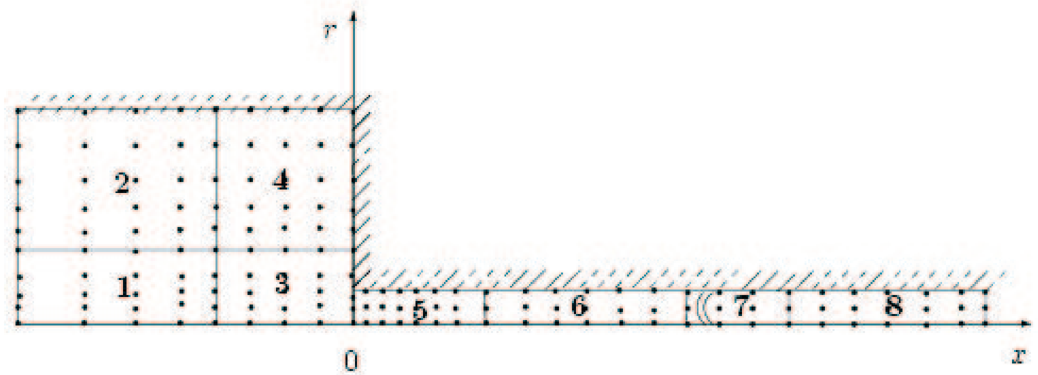


Figure 5.12: The steady state axisymmetric 4:1 contraction flow problem: A schematic domain decomposition of 8 sub-domains.

storage limitation, the iterative non-overlapping Domain Decomposition (DD) method is employed. Here, the analysis domain is divided into a number of sub-domains to reduce the rank (and then the condition number) of the coefficient matrices. In this chapter the DD technique is used only for the numerical solution of deterministic PDEs using the RBFN method.

The results presented just below correspond to the grid of 620 collocation points and using 8 sub-domains (Fig. 5.12). This problem is also solved with 4 and 12 sub-domains and the convergence behavior corresponding to the number of sub-domains is given in Table 5.3. Domain decomposition techniques and parallelization in conjunction with the RBFN-based element-free method are described in more detail in chapter 7.

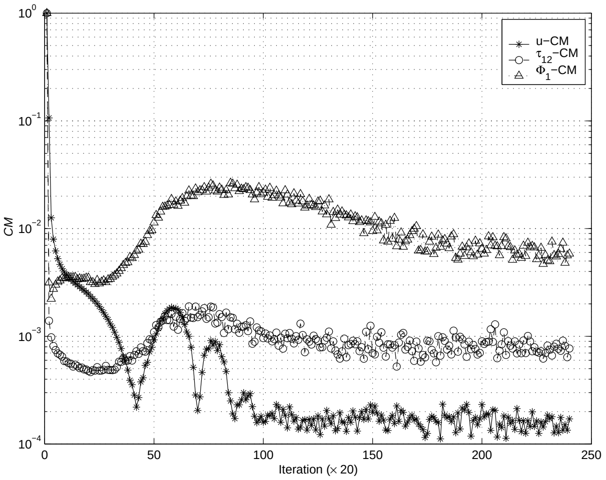


Figure 5.13: The steady state axisymmetric 4:1 contraction flow of the Hookean dumbbell model fluid: the convergence behavior of the velocity field CM_u, shear stress CM_τ and first normal stress difference CM_Φ with respect to the iteration number.

Results and discussion

The velocity and stress fields are obtained by averaging the results of 1000 iterations after attaining the desired convergence measures for the velocity field ($CM_u < 2.5e-4$), shear stress ($CM_\tau < 0.2e-2$) and the first normal stress difference ($CM_\Phi < 1.0e-2$), see Fig. 5.13. In this figure, the oscillations appear pronounced as the CMs are plotted on a base 10 logarithmic scale and the actual oscillations remain small and within the same order of magnitude in each case. Although small oscillations can generally be attributed to the nature of stochastic processes, it is not clear why there is an apparent minimum in the CM behaviour. However, this minimum is not too different from the final value of the CM (same order of magnitude). Fig. 5.14 shows the streamlines including the size of the vortex, which is in pretty good agreement with the results of Laso *et al.* (1995). Fig. 5.15 denotes the output velocity profile obtained from the present method in which the maximum velocity on the centreline is 4.9743 (while it is 5 in Laso *et al.* (1995)). For comparison with the results in Feigl *et al.* (1995), the polymer-contributed stress field (the shear stress τ_{xr} and the first

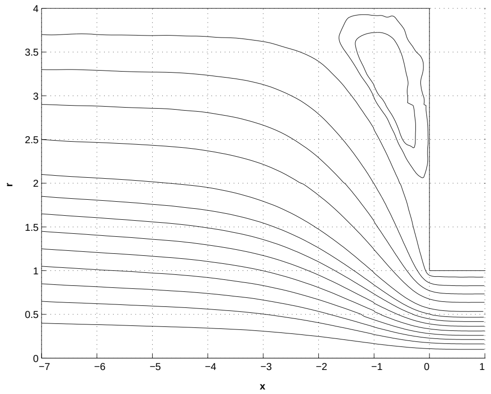


Figure 5.14: The steady state axisymmetric 4:1 contraction flow problem using the Hookean model: Streamlines of the velocity field.

normal stress difference $\tau_{xx} - \tau_{rr}$) is determined and plotted with respect to x and r along the length of the channel for three values of radius $r_1 = 0.05$ (near the centreline), $r_2 = 0.31$ and $r_3 = 0.945$ (near the wall of downstream). Figures 5.16, 5.17 and 5.18 are for the polymer shear stress and 5.19, 5.20 and 5.21 for the first polymer normal stress difference as functions of x $(-10 \leq x \leq 30)$ along the channel.

In general, the present results are in good agreement with the results depicted in figures 6-8 of Feigl *et al.* (1995) using CONNFFESSIT method. Several differences can be found as follows: for shear stress, in the region near the centreline of the section downstream of the abrupt contraction, the current result is higher than ones of Feigl *et al.* (1995), but nearer to the results from a macroscopic FEM, which were also presented in Feigl *et al.* (1995). The shear stress seems to oscillate less in the region near the centerline of the upstream section before the abrupt contraction but more on the downstream side (see Fig. 5.16 and figure 6 of Feigl *et al.* (1995)).

In this example, the domain is filled homogenously with 450000 dumbbells. Two cases of $N = 300000$ and 140000 dumbbells are also considered to assess the adequacy of the number of dumbbells used. Figs. 5.16-5.21 show that, owning to the use of the variance reduction technique and the overlapping EVSAs,

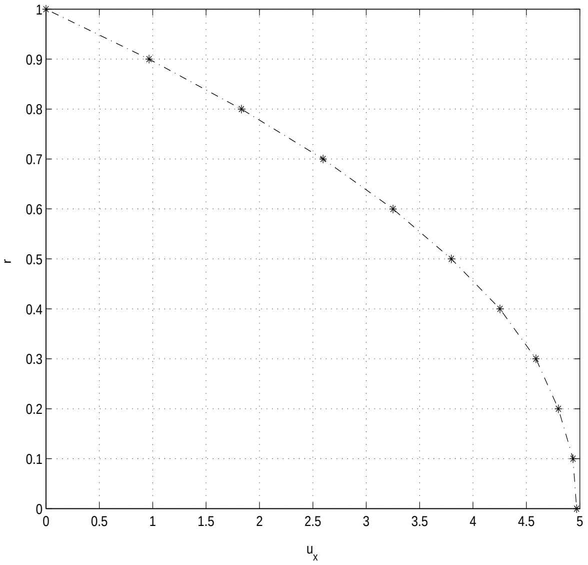


Figure 5.15: The steady state axisymmetric 4:1 contraction flow using the Hookean dumbbell model: the velocity profile at the outlet section.

a fewer number of dumbbells (300000) also gives good solutions for stresses. The deviation in stresses increases noticeably (Figs. 5.18 and 5.21) when using 140000 dumbbells. The Reynold and Deborah numbers associated with problem are given by $Re = \rho V_d r_d / \eta_o \approx 0.5$, $De = \frac{\lambda_H V_d}{r_d} \approx 5$, respectively.

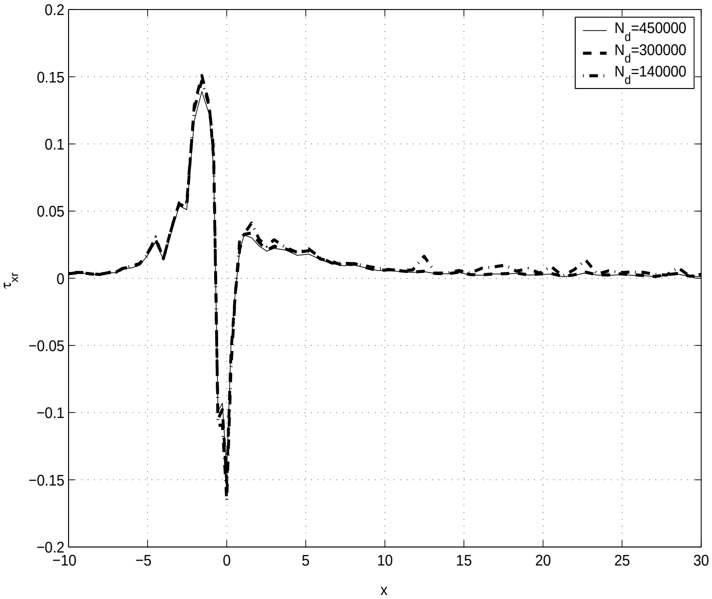


Figure 5.16: The steady state axisymmetric 4:1 contraction flow problem using the Hookean model: The polymer shear stress on the cylinder $r = 0.05$.

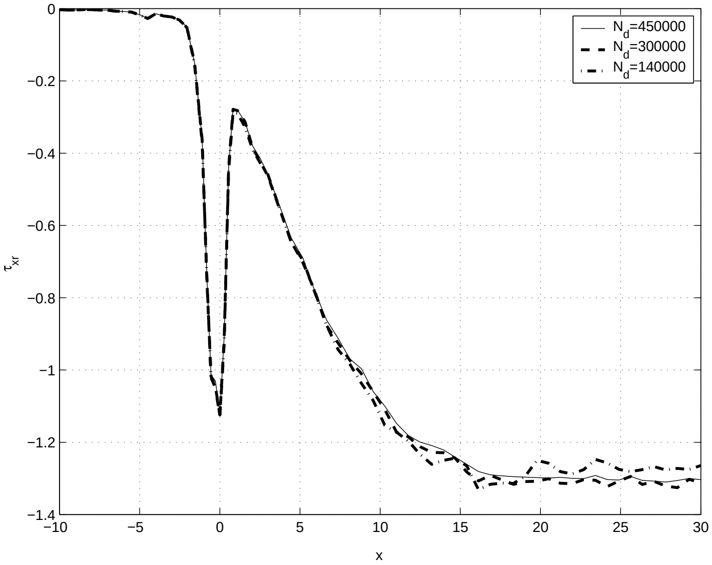


Figure 5.17: The steady state axisymmetric 4:1 contraction flow problem using the Hookean model: The polymer shear stress on the cylinder $r = 0.31$.

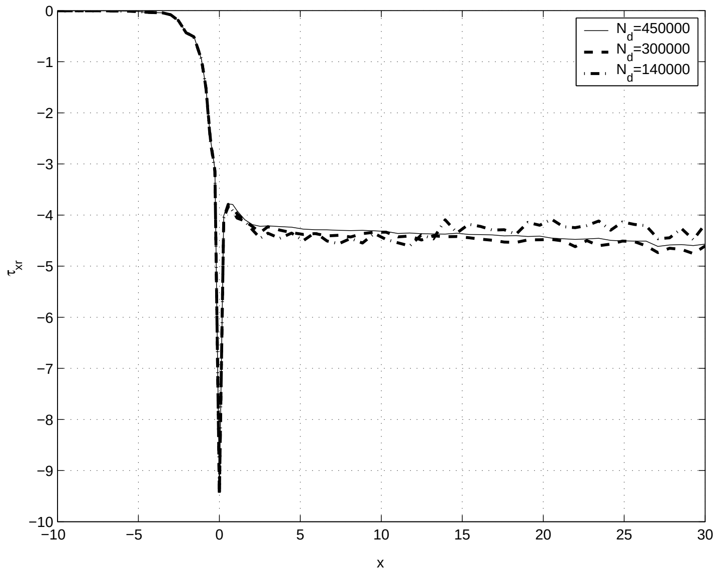


Figure 5.18: The steady state axisymmetric 4:1 contraction flow problem using the Hookean model: The first polymer normal stress difference on the cylinder $r = 0.945$.

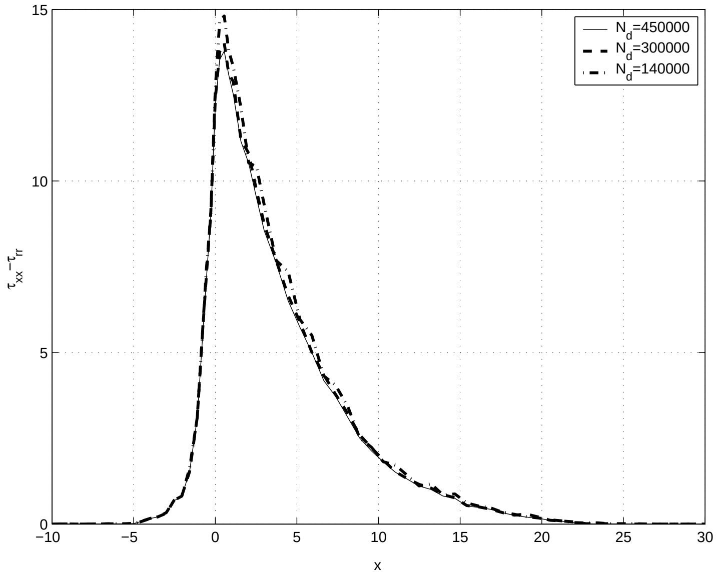


Figure 5.19: The steady state axisymmetric 4:1 contraction flow problem using the Hookean model: The first polymer normal stress difference on the cylinder $r = 0.05$.

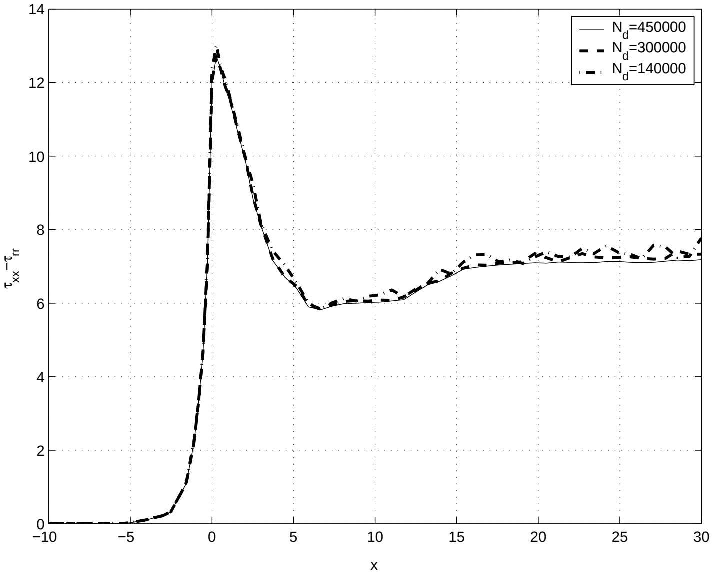


Figure 5.20: The steady state axisymmetric 4:1 contraction flow problem using the Hookean model: the polymer shear stresses on the cylinder $r = 0.31$ near the reentrant corner.

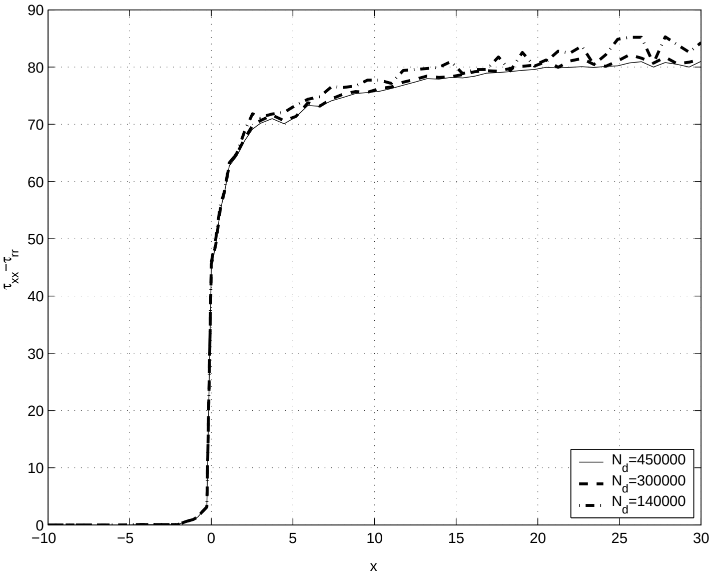


Figure 5.21: The steady state axisymmetric 4:1 contraction flow problem using the Hookean model: the polymer first normal stress difference on the cylinder $r = 0.945$ near the centre-line.

5.8 Concluding remarks

A new method for the computation of viscoelastic flows based on the combination of a RBFN mesh-free numerical procedure and a Lagrangian SST is demonstrated. The resultant micro-macro approach is able to solve several 2-D viscoelastic flow problems.

The present method adopt the micro-macro approach of the CONNFFESSIT idea (Feigl *et al.*, 1995; Laso *et al.*, 1997, 1999). However, the present method has several advantages. Firstly, it is a mesh-free numerical method where the governing PDEs are discretised simply by a set of random collocation points. Secondly, the approximation of the variables is achieved with global RBF and consequently it is also simple to generate effective volumes surrounding collocation points (non-overlapped or overlapped) for the purpose of determining polymer average stresses. The variance reduction technique and EVSAs allow for a reduction of the number of dumbbells, and hence the computation time is decreased. Thirdly, since the stress tensor is determined by the SST, the number of unknowns in the macroscopic component is significantly reduced. Furthermore, the pressure is dealt with by a penalty function technique, again reducing the number of unknowns. Smaller number of unknowns improves the condition number of the system, memory requirement and processing time. The use of TPS-RBFN-based approximation results in a very smooth global stress tensor. Finally, the inclusion of additional PBFs in the RBFNs improves remarkably the accuracy and the convergence rate of the method. Besides the advantages mentioned above, there exists several drawbacks associated with methods based on the Lagrangian point of view, such as the present method: (i) extra effort is required for both the particle tracking process and the stochastic simulation technique, (ii) the implementation of the variance reduction technique is more difficult. These drawbacks can be avoided by the use of the Brownian dynamics simulation from the Eulerian point view, namely BCFs, which will be presented in the next chapter.

Chapter 6

Element-free simulation of dilute polymeric flows using Brownian Configuration Fields

This chapter reports the CVFNNSS from the Eulerian point of view (Tran-Canh and Tran-Cong, 2003a). The present method is based on the combination of RBFN-based mesh-free method and Brownian configuration fields (BCFs) where the stress is computed from an ensemble of continuous configuration fields instead of convecting discrete particles, and the velocity field is determined by solving the conservation equations for mass and momentum with a truly mesh-free method based on RBFNs. Owing to nature of the BCFs, the control variate variance reduction method is installed easily in the method. The method is verified and its capability is demonstrated with the start-up planar Couette flow, the Poiseuille flow and the lid driven cavity flow of Hookean and FENE model materials.

6.1 Introduction

While the hybrid simulations using Brownian dynamics from the Lagrangian point of view, namely CONNFFESSIT, have been introduced as a robust method to solve the viscoelastic problems without the requirement of closed-form constitutive equations, the BCFs (Hulsen *et al.*, 1997) makes this mesoscopic approach easier in implementation and more efficient in performance. The BCF method, also called Eulerian CONNFFESSIT (Suen *et al.*, 2002), uses an ensemble of configuration fields which represent the internal degrees of freedom of the polymer molecules and bypasses the particle tracking process and hence the extra effort associated with this process. Many works have used successfully this approach (Fan *et al.*, 1998; Bonvin and Picasso, 1999; Somasi and Khomami, 2000) where stochastic simulation techniques (SST for the calculation of the stress tensor) are coupled with element-based methods (e.g. FEM for the solution of the governing equations such as the continuity and momentum equations) in a micro-macroscopic approach.

In general, as an alternative to element-based discretisation of PDEs, various finite point methods can be used in the so called meshless approach. Tran-Canh and Tran-Cong (2002b, 2003a) coupled successfully the RBFN-based meshless method with Lagrangian SST for the numerical solution of the start-up Couette and 2D viscoelastic fluid flows. In the macroscopic part, the discrete model is completely represented by a set of unstructured discrete collocation nodes in the analysis domain and on its boundary (i.e. there is no need to generate finite elements or define any topological connectivity) which is commonly referred to as truly meshless or mesh-free or element-free approach. In other words, the method can at least avoid the extra effort of meshing and re-meshing (if the problem requires) associated with the element type methods. However, effective volumes for stress averages (EVSA) can be flexibly generated around the collocation points to help determine the average polymer stress (see 5.3.3). The resultant element-free RBFN-SST method is Lagrangian as far as the microscopic part is concerned, and particle tracking could be inconvenient as described in chapter 5.

In this chapter, an Eulerian element-free RBFN-SST method is developed following the BCF idea. In the present method, the polymer contribution to stress at all collocation points is calculated using the BCF technique and then the continuity and momentum equations are solved using the RBFN-based method for the velocity field and pressure. The chapter is organized as follows: §6.2 is an outline of the scheme in which the governing PDEs and SDEs for the elastic dumbbell models are briefly reviewed. In §6.3, the RBFN-based numerical method for solving the conservation equations is briefly described, followed by numerical methods of the solution of BCF's. The associated variance reduction techniques are described for the SDE's for the Hookean and FENE dumbbell models. §6.4 presents the algorithm of the present scheme for viscoelastic flow problems, highlighting the macroscopic-microscopic interfaces of the method. Numerical examples are then discussed in §6.5, followed by a brief conclusion in §6.6.

6.2 Review on governing equations

The present work is concerned with the flow of dilute polymer solutions which are modelled as an incompressible suspension of non-interacting macromolecules in a Newtonian solvent. Under isothermal and steady state condition, an application of the penalty function method transforms the governing equations into

$$2\eta_N \boldsymbol{\nabla} \cdot \mathbf{D} - \rho(\mathbf{u} \cdot \boldsymbol{\nabla})\mathbf{u} + p_e \boldsymbol{\nabla}(\boldsymbol{\nabla} \cdot \mathbf{u}) = -\boldsymbol{\nabla} \cdot \boldsymbol{\tau}^p, \tag{6.1}$$

and the penalty equation is given by

$$p = -p_e(\boldsymbol{\nabla} \cdot \mathbf{u}), \tag{6.2}$$

subject to boundary conditions

$$\begin{aligned}\mathbf{u} &= u_o, \quad \mathbf{x} \in \Gamma_u,\\ \mathbf{n}\cdot\boldsymbol{\nabla}\mathbf{u} &= q_o, \quad \mathbf{x} \in \Gamma_t,\end{aligned}$$

where $\mathbf{u}$ denotes the velocity field; $\mathbf{n}$ is the unit vector outwardly normal to the boundary; $\mathbf{D}$ is the rate of strain tensor; η_N is the Newtonian solvent viscosity; $\boldsymbol{\tau} = 2\eta_N\mathbf{D} + \boldsymbol{\tau}^p$ is the extra stress; ρ is the fluid density; p_e is a sufficiently large penalty parameter (please see §5.2).

The microscopic method employs the Brownian dynamics simulation (or SST) to determine the polymer contribution to stress $\boldsymbol{\tau}^p$ via kinetic modelling. The kinetic theory-based models used here are the Hookean and FENE dumbbell models in which the configuration of a dumbbell satisfies a certain stochastic differential equation (SDE). In the Lagrangian approach, it is necessary to convect a large number of molecules through the domain under consideration, which requires extra effort associated with particle tracking. Hulsen *et al.* (1997) proposed a modified CONNFFESSIT method which overcomes these drawbacks. The method employs an ensemble of n_d continuous configuration fields $\mathbf{Q}(\mathbf{x}, t)$ with respect to space and time instead of convecting discrete connector vectors $\mathbf{Q}_i$'s. The main idea of this scheme is that after initiating n_d spatially uniform configuration fields $(n_d, \mathbf{Q})$ whose values are independently sampled from an equilibrium distribution function, the configuration fields are convected and deformed by the drift component (flow gradient, elastic retraction) and by the diffusion component (Brownian motion). This evolution of a configuration field satisfies the following SDE

$$d\mathbf{Q} = \left[-\mathbf{u}\cdot\boldsymbol{\nabla}\mathbf{Q} + \boldsymbol{\kappa}\cdot\mathbf{Q} - \frac{2}{\zeta}\mathbf{F}(\mathbf{Q})\right]dt + \sqrt{\frac{4k_BT}{\zeta}}d\mathbf{W}(t), \qquad (6.3)$$

where ζ is the friction coefficient between the dumbbell and the solvent; k_B is Boltzmann constant; T is the absolute temperature; $\mathbf{W}(t)$ is a 3-component vector which is a Wiener process with mean $\langle W_i(t)\rangle = 0$ and covariance $\langle W_i(t)W_j(t')\rangle$ $= \delta_{ij}\min(t, t')$ and accounts for the random displacement of the beads due to thermal motion; $\boldsymbol{\kappa}$ is the velocity gradient; $\mathbf{F}$ is the spring connector force between the two beads and depends on the model. Letting H be the spring constant, the connector force is given by

$$\begin{aligned} \mathbf{F} &= H\mathbf{Q}, & (6.4)\\ \mathbf{F} &= H\frac{\mathbf{Q}}{1-\left(\frac{\mathbf{Q}}{Q_o}\right)^2}, & (6.5) \end{aligned}$$

for the Hookean and the FENE dumbbell models, respectively, where Q_o is the maximum possible spring length. The configuration fields $(n_d, \mathbf{Q})$ are obtained by solving the SDE Eq (6.3). The term $\mathbf{u}(\mathbf{x}, t) \cdot \nabla\mathbf{Q}(\mathbf{x}, t)$ accounts for the convection of the configuration fields by the flow. It can be seen that the existence of the convective term in this Eulerian framework is completely equivalent to the particle tracking in the traditional Lagrangian CONNFFESSIT approach. Once the configuration fields are known, the stress can be determined as follows

$$\boldsymbol{\tau}^p = -n_d k_B T\mathbf{I} + n_d\langle\mathbf{Q}\cdot\mathbf{F}\rangle, \tag{6.6}$$

where n_d is the density of dumbbells; $\mathbf{I}$ is the identity tensor and $\mathbf{F}$ is the spring force. The configuration field $\mathbf{Q}$ is non-dimensionalised by $\sqrt{\frac{k_BT}{H}}$, and equation (6.3) becomes

$$\begin{aligned} d\mathbf{Q}'(t) = \left[-\mathbf{u}(\mathbf{x},t)\cdot\boldsymbol{\nabla}\mathbf{Q}'(\mathbf{x},t) + \boldsymbol{\kappa}(\mathbf{x},t)\cdot\mathbf{Q}'(\mathbf{x},t) - \frac{1}{2\lambda_H}\mathbf{F}^{'}(\mathbf{Q}^{'}(\mathbf{x},t))\right]dt \\ + \sqrt{\frac{1}{\lambda_H}}d\mathbf{W}(t), \quad (6.7) \end{aligned}$$

where $\mathbf{Q}^{'} = \mathbf{Q}[H/(k_BT)]^{1/2}$ is the dimensionless form of the configuration field vector $\mathbf{Q}$; $\lambda_H = \zeta/(4H)$ is the relaxation time of dumbbells; $b = \frac{HQ_o^2}{k_BT}$ is the square of the maximum possible extension of the dimensionless configuration field $\mathbf{Q}'$ and $\mathbf{F}'$ is the dimensionless spring force given by

$$\begin{aligned} \mathbf{F}' &= \mathbf{Q}', & (6.8)\\ \mathbf{F}' &= \frac{\mathbf{Q}'}{1-\frac{\mathbf{Q}'^2}{b}}, & (6.9) \end{aligned}$$

for the Hookean and FENE dumbbell models, respectively. For the sake of brevity, primes will be dropped in the following discussion.

6.3 Computational schemes

In this section, computational techniques are described for the numerical solution of the conservation equations (momentum and continuity equations) and the Brownian configuration fields, respectively. For the stochastic processes, a variance reduction technique is described, followed by a presentation of the overall algorithm.

6.3.1 RBFN-based element-free method for solving the momentum and continuity equations

An element-free method based on RBFNs for solving PDEs was developed and reported elsewhere (e.g. Tran-Canh and Tran-Cong, 2002b). Briefly, the method takes advantage of the fact that a smooth function can be approximated by a RBFN such as (Haykin, 1999; Golberg *et al.*, 1996)

$$f(\mathbf{x}) = \sum_{j=1}^{m} w^j h^j(\mathbf{x}) + \sum_{k=1}^{\bar{m}} \iota^k p^k(\mathbf{x}) = \mathbf{H}^T(\mathbf{x})\mathbf{w} + \mathbf{P}^T(\mathbf{x})\boldsymbol{\iota}, \tag{6.10}$$

$$\sum_{i=1}^{m} p^k(\mathbf{x}_i)\omega^i = 0, \quad k = 1, \cdots, \bar{m}, \tag{6.11}$$

where $w^j \in \mathbf{w}$ ($\mathbf{w}^T = [w^1 \quad w^2 \quad ... \quad w^m]$) and $\iota^k \in \boldsymbol{\iota}$ ($\boldsymbol{\iota}^T = [\iota^1 \quad \iota^2 \quad ... \quad \iota^{\bar{m}}]$) are the synaptic weights; h^j is the chosen radial basis function corresponding to the j^{th} RBF-neuron; p^k is the polynomial basis function corresponding to the k^{th} PBF-neuron; $m + \bar{m}$ is the total number of neurons. **H** and **P** are defined as follows

$$\begin{aligned} \mathbf{H}^T(\mathbf{x}) &= [h^1(\mathbf{x}) \quad h^2(\mathbf{x}) \quad ... \quad h^m(\mathbf{x})], & (6.12) \\ \mathbf{P}^T(\mathbf{x}) &= [p^1(\mathbf{x}) \quad p^2(\mathbf{x}) \quad ... \quad p^{\bar{m}}(\mathbf{x})]. & (6.13) \end{aligned}$$

Let n be the number of training collocation points $(\mathbf{x}_i, \hat{y}_i)$; $\mathbf{x}_i$ is the coordinate of the i^{th} collocation point and $\hat{y}_i$ is the desired value of function f at the collocation point $\mathbf{x}_i$.

RBFN procedure

As discussed in chapter 5, here collocation points are chosen to be the same as RBF centres, i.e. $m = n$, which are uniformly distributed in the physical domain. The unknown weights are found by minimizing an appropriate cost function given by

$$C(\mathbf{w}^{'}, \lambda) = \sum_{i=1}^{n} (\hat{y}_i - f(\mathbf{x}_i))^2 + \lambda \sum_{j=1}^{n} \tilde{w}^{(j)2}, \tag{6.14}$$

where λ is a global regularization parameter; $\tilde{w}^j \in \mathbf{w}^{'}$, then the partial derivatives of $f(\mathbf{x})$ can be calculated analytically as follows

$$\mathcal{L}f(\mathbf{x}) = \sum_{j=1}^{n} w^j \mathcal{L}h^j(\mathbf{x}) + \sum_{k=1}^{\bar{m}} \iota^k \mathcal{L}p^k(\mathbf{x}), \tag{6.15}$$

where $\mathcal{L}$ is a derivative operator

In particular, each variable in the momentum and continuity equations is approximated by an RBFN such as (6.10), and those equations are collocated at chosen points throughout the analysis domain, yielding the following sum square error (non-dimensional form) as follows

$$\begin{aligned} SSE = & \\ \sum_{\mathbf{i}\in\Omega} & \left\{ \alpha \left[\frac{\partial^2 u_1}{\partial x_1^2} + \frac{\partial^2 u_1}{\partial x_2^2} \right] - Re \left[u_1 \frac{\partial u_1}{\partial x_1} + u_2 \frac{\partial u_1}{\partial x_2} \right] + p_e \left[\frac{\partial^2 u_1}{\partial x_1^2} + \frac{\partial^2 u_1}{\partial x_1 \partial x_2} \right]_i + \phi_{1i} \right\}_{\mathbf{i}}^2 + \\ \sum_{\mathbf{i}\in\Omega} & \left\{ \alpha \left[\frac{\partial^2 u_2}{\partial x_1^2} + \frac{\partial^2 u_2}{\partial x_2^2} \right] - Re \left[u_1 \frac{\partial u_2}{\partial x_1} + u_2 \frac{\partial u_2}{\partial x_2} \right] + p_e \left[\frac{\partial^2 u_2}{\partial x_2^2} + \frac{\partial^2 u_2}{\partial x_1 \partial x_2} \right]_i + \phi_{2i} \right\}_{\mathbf{i}}^2 + \\ & \sum_{\mathbf{i}\in\Gamma_u} \{u_1 - u_o\}^2 + \sum_{\mathbf{i}\in\Gamma_t} \left\{ [n_1 \frac{\partial u_1}{\partial x_1} + n_2 \frac{\partial u_1}{\partial x_2}] - q_o \right\}_{\mathbf{i}}^2, \end{aligned} \tag{6.16}$$

where i denotes the i^{th} collocation point; $\alpha = \frac{\eta_N}{\eta_o}$; $\eta_o = \eta_N + \eta_p$, η_p is the polymer viscosity; $Re = \frac{\rho V a}{\eta_o}$, V and a are characteristic velocity and length, respectively; $\phi_{1i} = \frac{\partial \tau_{11}^p}{\partial x_1}(\mathbf{x}_i) + \frac{\partial \tau_{21}^p}{\partial x_2}(\mathbf{x}_i)$; $\phi_{2i} = \frac{\partial \tau_{12}^p}{\partial x_1}(\mathbf{x}_i) + \frac{\partial \tau_{22}^p}{\partial x_2}(\mathbf{x}_i)$. The stresses are scaled by $\eta_o V/a$. Applying the general linear least-squares principle to (6.16) (taking into account (6.11)), a system of linear algebraic equations of the unknown weights is obtained as follows

$$\mathbf{G}\mathbf{w}' = \mathbf{B}^T\hat{\mathbf{y}}, \tag{6.17}$$

where $\mathbf{G}$ is the variance matrix and $\mathbf{B}$ is the design matrix defined in §5.5.2; $\mathbf{w}'$ is the vector of all weights; $\hat{\mathbf{y}}$ is the vector of known values. In the present work, the RBFN-based least squares method did not encounter any instability. Furthermore, several recent works (Fasshauer, 1996; Franke and Schaback, 1998 and Kansa and Hon, 2000) have pointed out that the RBFN-based meshfree methods are stable and do not require any stabilizing schemes. In this chapter, the TPS of Duchon (1976) is chosen as the RBF which was presented in chapter 5.

6.3.2 Numerical simulation of the configuration fields

In this section, the second-order semi-implicit predictor-corrector scheme is applied in solving the governing Eulerian SDE of Brownian configuration fields (6.7). Here, the time discretisation of the elastic dumbbell configuration fields is described only briefly for the FENE model. The technique consists of two steps as follows

(i) The predictor step

For a dumbbell, let $\mathbf{Q}_n$, $\mathbf{Q}_{n+1}$ be an approximation of $\mathbf{Q}(t)$ at time steps t_n and t_{n+1}, respectively. Using a fixed time stepsize Δt for the stochastic process, the predicted BCF $\mathbf{Q}^*_{n+1}$at the time step t_{n+1} is explicitly determined as follows:

$$\mathbf{Q}^*_{n+1} = \mathbf{Q}_n - \left(\mathbf{u}_n \cdot \nabla\mathbf{Q}_n - \boldsymbol{\kappa}_n \cdot \mathbf{Q}_n - \frac{\mathbf{Q}_n}{2\lambda_H\left(1 - \frac{\mathbf{Q}_n^2}{b}\right)}\right)\Delta t + \sqrt{\frac{\Delta t}{\lambda_H}}\mathbf{W}_n, \tag{6.18}$$

where the components of the random vector $\mathbf{W}_n$ are independent Gaussian variables with zero mean and Δt variance. Since $\mathbf{W}_n$ depends on time only, it affects the configuration fields in a spatially uniform way and hence the gradient of the configuration fields are well defined as smooth functions with respect to spatial coordinates (Hulsen *et al.*, 1997). The gradient of the configuration and velocity, $\boldsymbol{\nabla}\mathbf{Q}_n$ and $\boldsymbol{\kappa}_n$, at time-step t_n are determined by calculating directly the derivatives of the approximant of $\mathbf{Q}_n$ and $\mathbf{u}_n$ (TPS-RBFNs), as shown in Eq (6.15). The updated configuration fields $\mathbf{Q}^*_{n+1}$ are employed to estimate the polymer contribution to the predicted stress $\boldsymbol{\tau}^*_{n+1}$, according to (6.6), which is in turn used to get the solutions of the predicted velocity at time t_{n+1} by solving Eq (6.1). The predicted velocity fields $\mathbf{u}^*_{n+1}$, also approximated by TPS-RBFNs, and associated gradients are employed in the corrector step.

(ii) The corrector step

$$\begin{aligned}\left(1+\frac{\Delta t}{4\lambda_H\left(1-\frac{\mathbf{Q}_{n+1}}{b}\right)}\right)\mathbf{Q}_{n+1} = \mathbf{Q}_n + \frac{1}{2}\Bigg(&-\mathbf{u}_n\cdot\nabla\mathbf{Q}_n-\mathbf{u}^*_{n+1}\cdot\nabla\mathbf{Q}^*_{n+1}\\ &+\boldsymbol{\kappa}_n\cdot\mathbf{Q}_n+\boldsymbol{\kappa}^*_{n+1}\cdot\mathbf{Q}^*_{n+1}-\frac{\mathbf{Q}_n}{2\lambda_H(\left(1-\frac{\mathbf{Q}_n^2}{b}\right))}\Bigg)\Delta t+\sqrt{\frac{\Delta t}{\lambda_H}}\mathbf{W}_n. \qquad (6.19)\end{aligned}$$

Eq (6.19) leads to a unique cubic equation for $|\mathbf{Q}_{n+1}|$ of which admissible solutions are those that satisfy $0 \leq |\mathbf{Q}_{i+1}| < \sqrt{b}$ (Öttinger, 1996). It is noted that the gradients of the configuration and velocity on the RHS of (6.19) and (6.18) are determined by calculating directly the derivatives of their TPS-RBFN approximant as shown in Eq (6.15) and the random numbers $\mathbf{W}_n$ in the two predictor and corrector steps are the same.

The polymer stress tensor is then determined by the average of the configuration fields evaluated at each collocation point and given by Kramers' expression as follows (Bird *et al.*, 1987b; Öttinger, 1996):

$$\begin{aligned}\boldsymbol{\tau}^p &= -n_d k_B T\left(\langle\mathbf{Q}\mathbf{Q}\rangle-\mathbf{I}\right), &(6.20)\\ \boldsymbol{\tau}^p &= -n_d k_B T\left(\left\langle\frac{\mathbf{Q}\mathbf{Q}}{1-\frac{\mathbf{Q}^2}{b}}\right\rangle-\mathbf{I}\right), &(6.21)\end{aligned}$$

for the Hookean and FENE dumbbell models, respectively.

6.3.3 Control variate scheme for variance reduction in BCFs simulation

Owing to the Eulerian nature of the BCF scheme, the implementation of the the control variate techniques is achieved easily in the present approach. In this chapter, the implementation of the control variate method in conjunction with BCFs is presented for the Hookean and FENE dumbbell models as follows: at each collocation point, n_d dumbbells are assigned and numbered from $i = 1..n_d$ where dumbbells having the same index in the whole analysis domain have the same random number. Here, for illustration, this technique is employed for the numerical calculation of the polymer contribution to stress (6.20)-(6.21) where the expectations of random variables $\mathbf{QQ}$ and $\frac{\mathbf{QQ}}{1-\frac{\mathbf{Q}^2}{b}}$ are required, respectively, for the Hookean and FENE models. At each time t and position $\mathbf{x}$, let $\bar{\mathbf{Q}}(\mathbf{x}, t)$ be the control variate corresponding to the configuration field $\mathbf{Q}(\mathbf{x}, t)$. The variance reduction method is carried out by splitting the expectation above as follows (Bonvin and Picasso, 1999)

$$\begin{aligned} \langle \mathbf{QQ} \rangle &= \langle \bar{\mathbf{Q}}\bar{\mathbf{Q}} \rangle + \langle \mathbf{QQ} - \bar{\mathbf{Q}}\bar{\mathbf{Q}} \rangle , && (6.22) \\ \left\langle \frac{\mathbf{QQ}}{1-\frac{\mathbf{Q}^2}{b}} \right\rangle &= \left\langle \frac{\bar{\mathbf{Q}}\bar{\mathbf{Q}}}{1-\frac{\bar{\mathbf{Q}}^2}{b}} \right\rangle + \left\langle \frac{\mathbf{QQ}}{1-\frac{\mathbf{Q}^2}{b}} - \frac{\bar{\mathbf{Q}}\bar{\mathbf{Q}}}{1-\frac{\bar{\mathbf{Q}}^2}{b}} \right\rangle . && (6.23) \end{aligned}$$

When $\bar{\mathbf{Q}} = 0$ there is no variance reduction. From (6.22)-(6.23), the polymer stress tensors (6.20)-(6.21) are rewritten as follows

$$\begin{aligned} \boldsymbol{\tau}^p &= -n_d k_B T \left(\langle \mathbf{QQ} - \bar{\mathbf{Q}}\bar{\mathbf{Q}} \rangle \right) + \bar{\boldsymbol{\tau}}^p_{hook}, && (6.24) \\ \boldsymbol{\tau}^p &= -n_d k_B T \left(\left\langle \frac{\mathbf{QQ}}{1-\frac{\mathbf{Q}^2}{b}} - \frac{\bar{\mathbf{Q}}\bar{\mathbf{Q}}}{1-\frac{\bar{\mathbf{Q}}^2}{b}} \right\rangle \right) + \bar{\boldsymbol{\tau}}^p_{fene}, && (6.25) \end{aligned}$$

where $\bar{\boldsymbol{\tau}}^P_{hook}$ and $\bar{\boldsymbol{\tau}}^p_{fene}$ are given by

$$\begin{aligned} \bar{\boldsymbol{\tau}}^p_{hook} &= -n_d k_B T \left(\langle \bar{\mathbf{Q}}\bar{\mathbf{Q}} \rangle - \mathbf{I} \right) , && (6.26) \\ \bar{\boldsymbol{\tau}}^p_{fene} &= -n_d k_B T \left(\left\langle \frac{\bar{\mathbf{Q}}\bar{\mathbf{Q}}}{1-\frac{\bar{\mathbf{Q}}^2}{b}} \right\rangle - \mathbf{I} \right) . && (6.27) \end{aligned}$$

The first terms of the RHS of (6.24) and (6.25) are calculated by using Brownian dynamic simulations and the second terms are determined in a deterministic way. In the present work, since $\bar{\mathbf{Q}}$'s are estimated at equilibrium configuration both $\bar{\boldsymbol{\tau}}_{hook}$ and $\bar{\boldsymbol{\tau}}_{fene}$ are zero and the configuration vectors $\bar{\mathbf{Q}}$'s satisfy the following SDE

$$d\bar{\mathbf{Q}} = -\frac{\bar{\mathbf{F}}}{2\lambda_H}dt + \sqrt{\frac{1}{\lambda_H}}d\mathbf{W}(t), \tag{6.28}$$

where $\bar{\mathbf{F}}$ is given by

$$\begin{aligned} \bar{\mathbf{F}} &= \bar{\mathbf{Q}}, & (6.29)\\ \bar{\mathbf{F}} &= \frac{\bar{\mathbf{Q}}}{1-\frac{\bar{\mathbf{Q}}^2}{b}}, & (6.30)\end{aligned}$$

for Hookean and FENE dumbbell models, respectively. The polymer stress tensors (6.24)-(6.25) are rewritten as follows

$$\begin{aligned} \boldsymbol{\tau}^p &= -n_d k_B T\left(\langle \mathbf{QQ} - \bar{\mathbf{Q}}\bar{\mathbf{Q}}\rangle\right), & (6.31)\\ \boldsymbol{\tau}^p &= -n_d k_B T\left(\left\langle \frac{\mathbf{QQ}}{1-\frac{\mathbf{Q}^2}{b}} - \frac{\bar{\mathbf{Q}}\bar{\mathbf{Q}}}{1-\frac{\bar{\mathbf{Q}}^2}{b}}\right\rangle\right). & (6.32)\end{aligned}$$

6.4 Algorithm of the present method

In general, the overall approach can now be described in a detailed algorithm (see Figs. 6.1 and 6.2 for flowcharts) as follows:

(a) Generate a set of collocation points and start with an initial velocity for the first iteration (zero in the present work) along with the boundary conditions of problem;

(b) Assign n_d dumbbells to each collocation point. These dumbbells are numbered from $i = 1$ to n_d. All dumbbells having the same index constitute a configuration. Hence there is an ensemble of n_d configuration fields $\mathbf{Q}_i$ ($i = 1..n_d$). Initially, the polymer configuration fields are spatially uniform and their values are independently sampled from the known equilibrium distribution function which is a 3-D Gaussian distribution with zero mean and unit covariance (Bird *et al.*, 1987b; Öttinger, 1996). Since all the dumbbells having the same index receive the same random numbers, there is a strong correlation between dumbbells in a configuration. The control variates $\bar{\mathbf{Q}}_i$'s associated with the configuration fields $\mathbf{Q}_i$'s are created as described in §6.3.3;

(c) Calculate velocity gradient fields directly by differentiating the RBFNs that approximate the velocity fields;

(d) Calculate the polymer configuration fields using the method described in §6.3.2. To ensure strong correlation within a configuration field, all the dumbbells of the same index have the same random numbers. For each configuration field $\mathbf{Q}$, a corresponding control variate $\bar{\mathbf{Q}}$ is determined according to the procedure described in §6.3.3. In this work, while the time discretisation of the BCF is carried out by a predictor-corrector scheme, the control variates which are governed by Eq (6.28) is estimated by Euler method;

(e) Determine the polymer contribution to stress by taking the ensemble average of the polymer configurations at each collocation point using (6.24) and (6.25) for the Hookean and the FENE dumbbell models, respectively. Impose the stress boundary conditions at the collocation points located on the boundary;

(f) The stress is then approximated globally by TPS-RBF networks which are the ultimate description of the stress field;

(g) With the stress field just obtained, solve the set of conservation equations for the new velocity field using the RBFN-based mesh-free method as described in §6.3.1;

(h) Terminate the simulation when either the desired time or convergence is reached. The latter is determined by a convergence measure for either the velocity field or the stress field, which is defined for the velocity field by

$$CM = \sqrt{\frac{\sum_1^N \sum_{i=1}^d (u_i^n - u_i^{n-1})^2}{\sum_1^N \sum_{i=1}^d (u_i^n)^2}} < tol, \tag{6.33}$$

where d is the number of dimension (2 in the present work); tol is a preset tolerance; u_i is the i component of the velocity at a collocation point; N is the total number of collocation points and n is the iteration number. Convergence is also checked for the shear stress and the first normal stress difference;

(i) Return to step (d) for the next time level of the microscopic process.

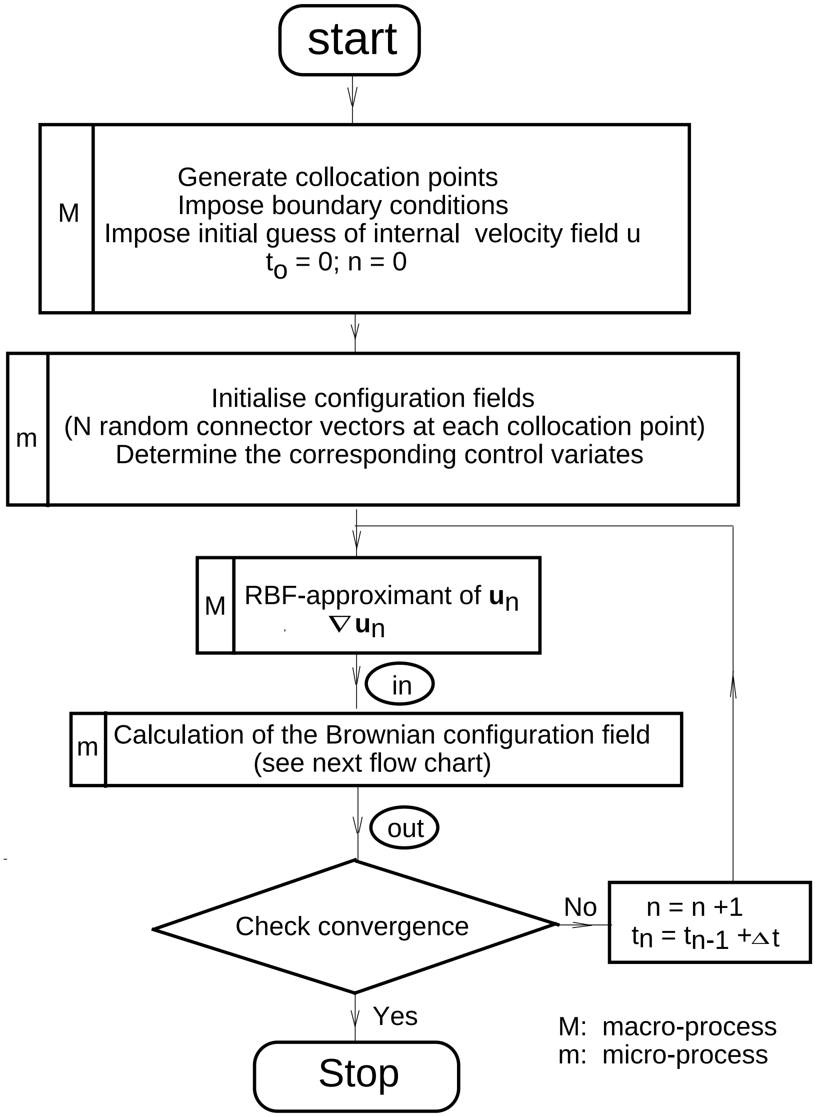

start
M
Generate collocation points
Impose boundary conditions
Impose initial guess of internal velocity field u
t_o = 0; n = 0
m
Initialise configuration fields
(N random connector vectors at each collocation point)
Determine the corresponding control variates
M
RBF-approximant of u_n
∇u_n
in
m
Calculation of the Brownian configuration field
(see next flow chart)
out
Check convergence
No
n = n +1
t_n = t_n-1 +△t
Yes
Stop
M: macro-process
m: micro-process

Figure 6.1: General flow-chart of the present meshless RBFN-SST method. See Fig. 6.2 for details of the calculation of the BCFs.

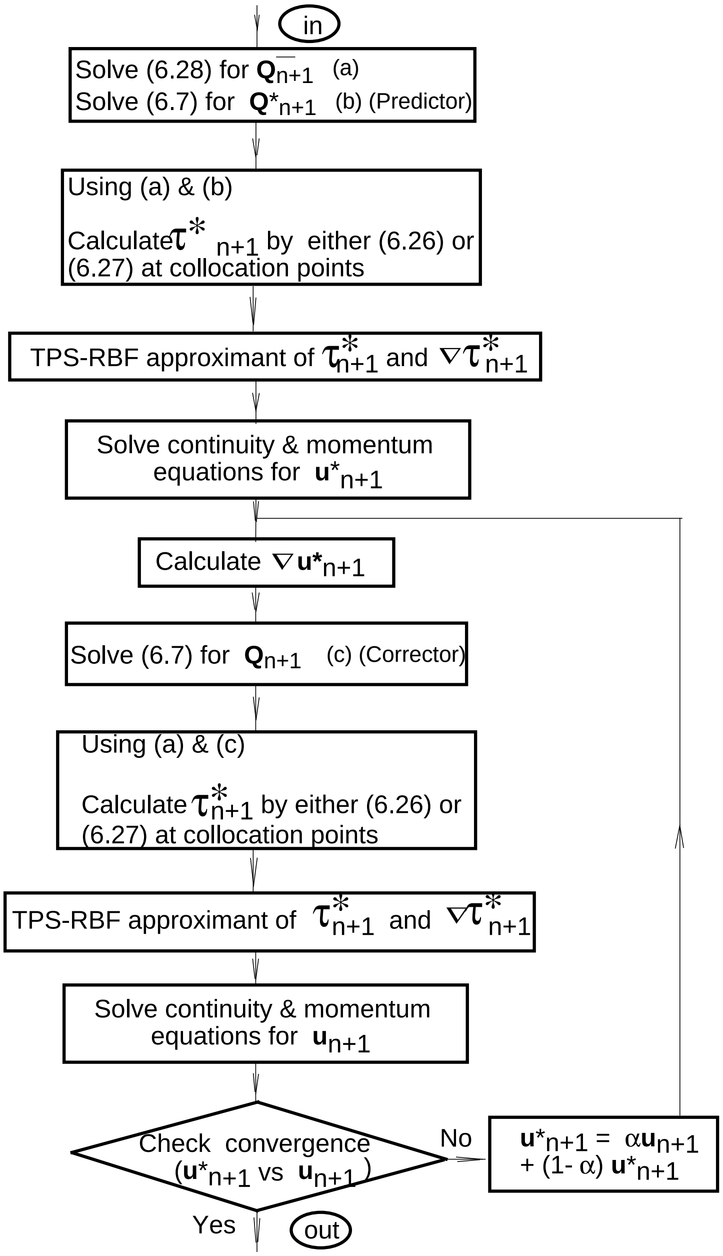

in
Solve (6.28) for $\overline{\mathbf{Q}}_{n+1}$ (a)
Solve (6.7) for $\mathbf{Q}^*_{n+1}$ (b) (Predictor)
Using (a) & (b)
Calculate τ^*_{n+1} by either (6.26) or (6.27) at collocation points
TPS-RBF approximant of τ^*_{n+1} and $\nabla\tau^*_{n+1}$
Solve continuity & momentum equations for $\mathbf{u}^*_{n+1}$
Calculate $\nabla \mathbf{u}^*_{n+1}$
Solve (6.7) for $\mathbf{Q}_{n+1}$ (c) (Corrector)
Using (a) & (c)
Calculate τ^*_{n+1} by either (6.26) or (6.27) at collocation points
TPS-RBF approximant of τ^*_{n+1} and $\nabla\tau^*_{n+1}$
Solve continuity & momentum equations for $\mathbf{u}_{n+1}$
Check convergence ($\mathbf{u}^*_{n+1}$ vs $\mathbf{u}_{n+1}$)
No
$\mathbf{u}^*_{n+1} = \alpha\mathbf{u}_{n+1} + (1-\alpha)\,\mathbf{u}^*_{n+1}$
Yes
out

Figure 6.2: Details of the calculation of the BCFs using semi implicit predictor corrector method. This flowchart represents part of the overall flowchart shown in Fig. 6.1.

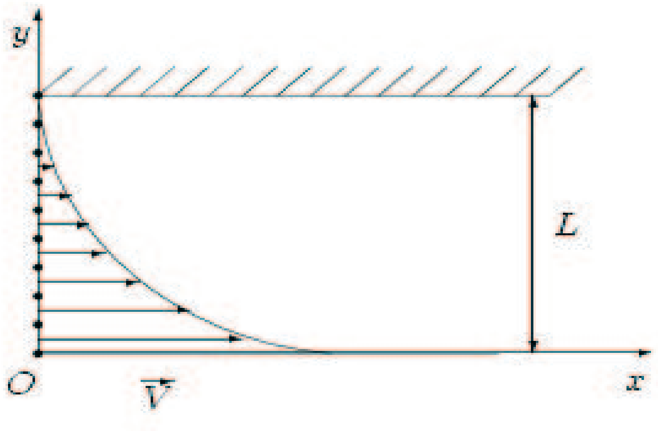


Figure 6.3: The start-up planar Couette flow problem: the bottom plate moves with a constant velocity $V = 1$, the top plate is fixed; no-slip boundary conditions apply at the fluid-solid interfaces. The collocation point distribution is only schematic.

6.5 Numerical examples

The present method is verified with the simulation of the start-up planar Couette and steady state planar Poiseuille flows of Hookean and FENE model fluids. The capability of the method is then demonstrated with the simulation of the lid driven cavity flow of the Hookean model fluid. For all examples, the criterion for convergence is $tol = 10^{-4}$ applied to the velocity field.

6.5.1 Start-up planar Couette flow

This problem was earlier studied by Mochimaru (1983) for the FENE-P model, by Laso and Öttinger (1993) and Tran-Canh and Tran-Cong (2002b) for the FENE and FENE-P models, and it is used here to verify the present method. The problem is defined in Fig. 6.3 and the chosen physical parameters are $\eta_o = \eta_N + \eta_p = 1$, $\rho = 1.2757$, $\quad \lambda_H = 49.62$, $\quad b = 50$, $\eta_N = 0.0521$, $\Delta t = 10^{-2}$ (Mochimaru, 1983; Laso and Öttinger, 1993).

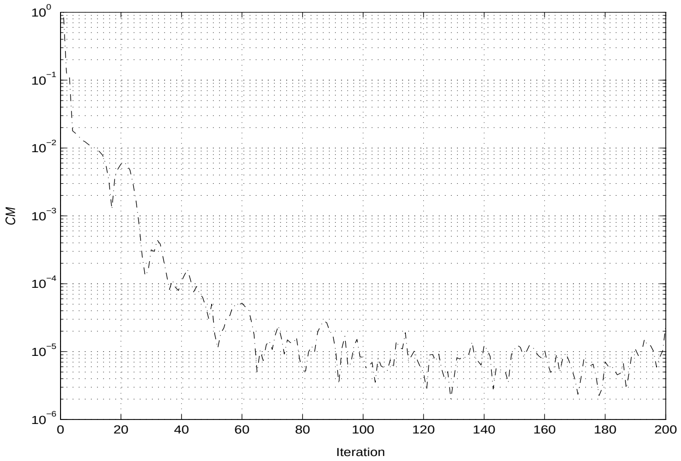


Figure 6.4: The start-up planar Couette flow problem using the FENE model: the velocity convergence rate. The parameters of the problem are number of collocation points = 25, the number of dumbbells at each collocation point = 1000, $\lambda_H = 49.62$, $b = 50$, $\eta_N = 0.0521$ and $\Delta t = 10^{-2}$.

To ensure that the centre density is adequate, three levels of discretisation are used, namely $n = 17$, $n = 23$ and $n = 25$, and the results show that the solutions obtained do not differ significantly. Only the results corresponding to $n = 25$ are presented here. The analysis is carried out for the FENE dumbbell model where the configuration fields are produced with one thousand dumbbells at each collocation point and the velocity convergence is shown in Fig. 6.4. The control variate is calculated at the equilibrium state. The simulation is continued for $t \geq 0$ until the flow reaches the steady state. Fig. 6.5 describes the evolution of the velocity at four locations $y = 0.2$, $y = 0.4$, $y = 0.6$ and $y = 0.8$ and shows that the velocity overshoot occurs sooner in fluid layers nearer to the moving wall. Fig. 6.6 depicts the evolution of the velocity profile with respect to the coordinate y, which confirms that velocity undershoot is insignificant in comparison with overshoot. Figs. 6.7 and 6.8 describe the evolution of the shear stress and the first normal stress difference, respectively at locations $y = 0.2$, $y = 0.4$, $y = 0.6$ and $y = 0.8$. The present result is a close match with the results of CONNFFESSIT (Laso and Öttinger, 1993). It is notable that the quality of convergence is better than that achieved with the Lagrangian CVFNNSS method reported in chapter 4.

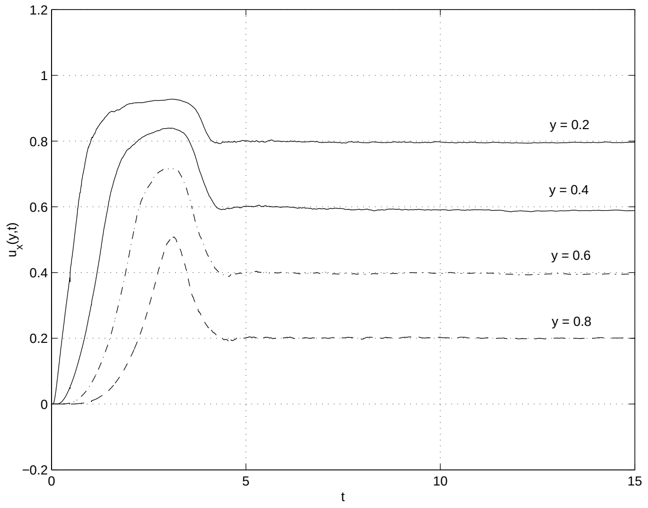


Figure 6.5: The start-up planar Couette flow problem using the FENE model: the time evolution of velocity at locations $y = 0.2$, $y = 0.4$, $y = 0.6$ and $y = 0.8$. The parameters of the problem are number of collocation points = 25, the number of dumbbells at each collocation point = 1000, $\lambda_H = 49.62$, $b = 50$, $\eta_N = 0.0521$ and $\Delta t = 10^{-2}$.

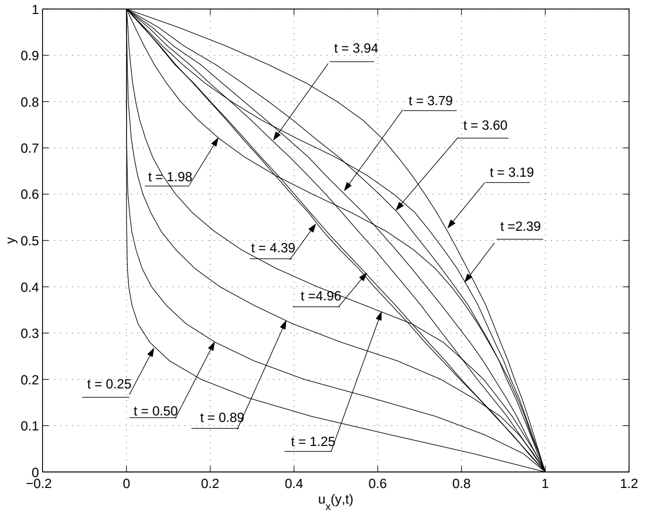


Figure 6.6: The start-up planar Couette flow problem using the FENE dumbbell model: the velocity profile with respect to location y at different times. The parameters are the same as in Fig. 6.5.

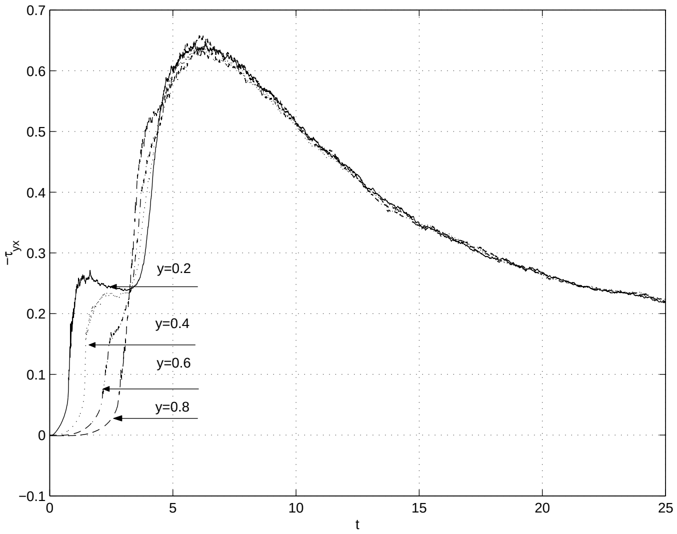


Figure 6.7: The start-up planar Couette flow problem using the FENE dumbbell model: the evolution of shear stress at locations $y = 0.2$, $y = 0.4$, $y = 0.6$, $y = 0.8$ with respect to time. The parameters are the same as shown in Fig. 6.5.

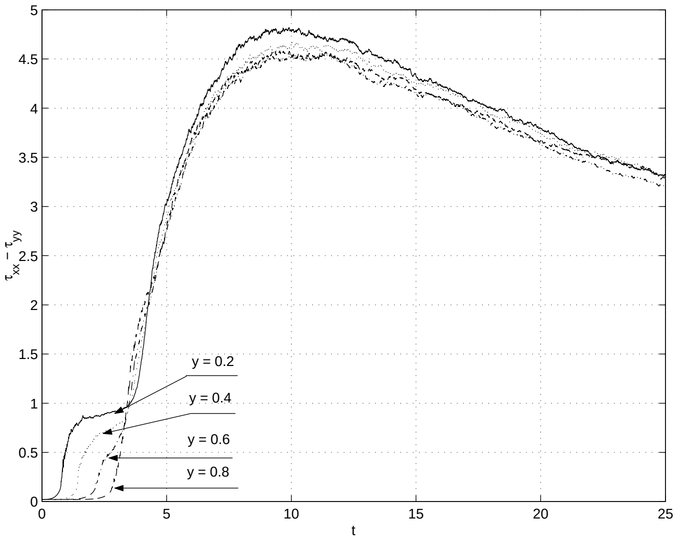


Figure 6.8: The start-up planar Couette flow problem using the FENE dumbbell model: the evolution of the first normal stress difference at the locations $y = 0.2$, $y = 0.4$, $y = 0.6$, $y = 0.8$ with respect to time. The parameters are the same as in Fig. 6.5.

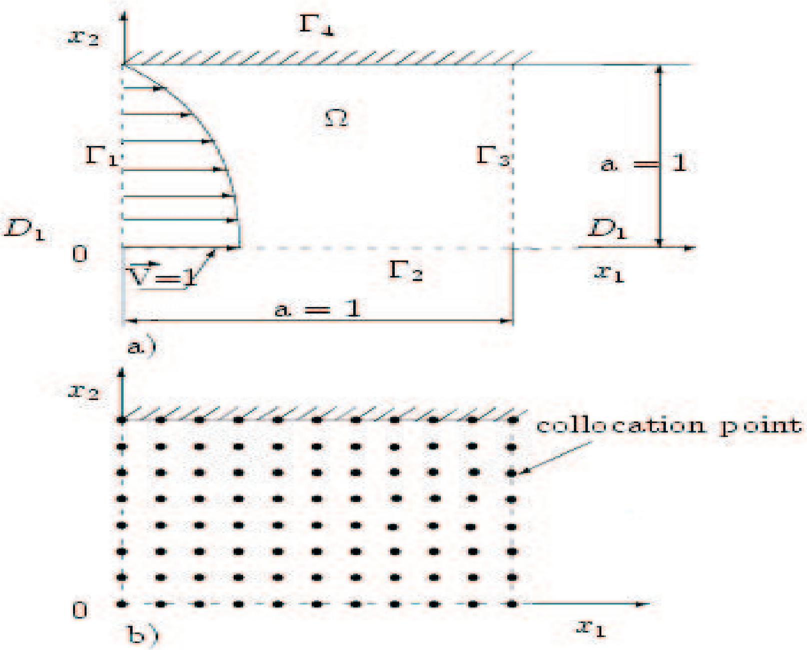


Figure 6.9: a) The planar Poiseuille flow problem with parabolic inlet velocity profile; non-slip boundary conditions applied at the fluid-solid interfaces. b) The collocation point distribution is only schematic.

6.5.2 The steady state planar Poiseuille flow

The planar creeping Poiseuille problem and coordinate system are described in Fig. 6.9a where only half of the fluid domain needs to be considered, owing to symmetry. The geometry and fluid parameters are the same as in §5.7.1. Using two collocation densities, namely 15 × 15 and 25 × 25, whose schematic distribution is shown in Fig. 6.9b, 1000 dumbbells are assigned at each collocation point. In this example, two models, namely Hookean and FENE, are considered. The boundary conditions were described in §5.7.1.

Sum square error

The expression of sum square error (6.16) for the creeping planar Poiseuille flows is given by

$$
\begin{aligned}
SSE = \sum_{\mathbf{x}_i\in\Omega}\left(\frac{\partial u_1}{\partial x_1}(\mathbf{x}_i)+\frac{\partial u_2}{\partial x_2}(\mathbf{x}_i)\right)^2 + \\
\sum_{\mathbf{x}_i\in\Omega}\left(\alpha\left[\frac{\partial^2 u_1}{\partial x_1^2}(\mathbf{x}_i)+\frac{\partial^2 u_1}{\partial x_2^2}(\mathbf{x}_i)\right]-\frac{\partial p}{\partial x_1}(\mathbf{x}_i)+\phi_1(\mathbf{x}_i)\right)^2 + \\
\sum_{\mathbf{x}_i\in\Omega}\left(\alpha\left[\frac{\partial^2 u_2}{\partial x_1^2}(\mathbf{x}_i)+\frac{\partial^2 u_2}{\partial x_2^2}(\mathbf{x}_i)\right]-\frac{\partial p}{\partial x_2}(\mathbf{x}_i)+\phi_2(\mathbf{x}_i)\right)^2 + \\
\sum_{\mathbf{x}_i\in\Gamma_1}\{u_1(\mathbf{x}_i)-u_p\}^2+\sum_{\mathbf{x}_i\in\Gamma_1}u_2^2(\mathbf{x}_i)+\sum_{\mathbf{x}_i\in\Gamma_2}u_2^2(\mathbf{x}_i)+\sum_{\mathbf{x}_i\in\Gamma_2}\left\{\frac{\partial u_1}{\partial x_2}(\mathbf{x}_i)\right\}^2 + \\
\sum_{\mathbf{x}_i\in\Gamma_3}u_2^2(\mathbf{x}_i)+\sum_{\mathbf{x}_i\in\Gamma_4}u_1^2(\mathbf{x}_i)+\sum_{\mathbf{x}_i\in\Gamma_4}u_2^2(\mathbf{x}_i), \quad (6.34)
\end{aligned}
$$

where $\phi_1(\mathbf{x}_i)=\frac{\partial\tau_{11}^p}{\partial x_1}(\mathbf{x}_i)+\frac{\partial\tau_{21}^p}{\partial x_2}(\mathbf{x}_i)$; $\phi_2(\mathbf{x}_i)=\frac{\partial\tau_{12}^p}{\partial x_1}(\mathbf{x}_i)+\frac{\partial\tau_{22}^p}{\partial x_2}(\mathbf{x}_i)$; u_p is the inlet velocity profile given in section §5.7.1.

Results and discussion

The solutions obtained for the velocity field, shear stress and the first normal stress difference are the averages of the last 200 iterations after reaching the steady state. For the Hookean dumbbell model, the parabolic velocity profile is accurately recovered in the downstream region as expected. Fig. 6.10 and 6.11 show the polymer shear stress and the first normal stress difference on the middle plane $x_1 = 0.5$ corresponding to the two collocation densities 15×15 and 25×25. The results are in very good agreement with the analytical solution given by Eq (5.43). Fig 6.10 also shows the polymer shear stress and the first normal stress difference at several steps after convergence (steps 120, 121, 122) which depicts small oscillation in steady state solutions as iteration goes on. Such oscillation has its origin in stochastic nature of the microscopic stress calculation, and therefore the final result is obtained by averaging a large number of these ‘steady state’ solutions. If different random numbers are generated for the initial configuration, different trajectories of velocity and stresses will be experienced but the final average results do not change, which is a characteristic

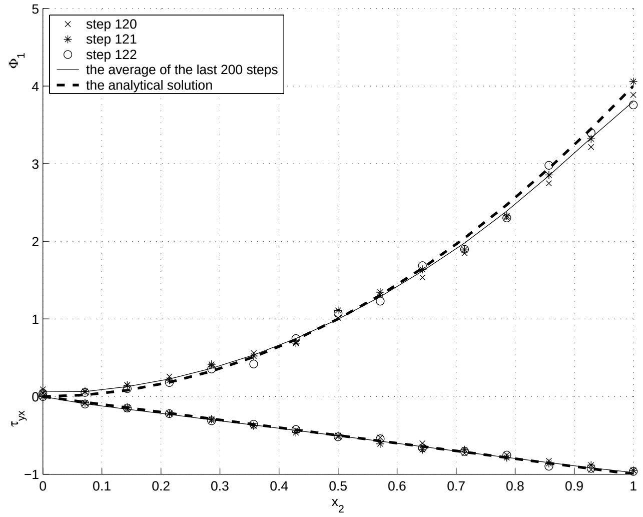


Figure 6.10: The steady state planar Poiseuille flow problem using the Hookean dumbbell (Oldroyd-B) model: the polymer shear stress and the first normal stress difference on the middle plane $x_1 = 0.5$ with respect to x_2 are denoted by '×' for the step 121, 'o' for step 122, solid line for step 123, solid line for the average of the last 200 steps and dashed line for the analytical solution, respectively.

of stochastic simulations (Öttinger, 1996). Furthermore, the error observed in Fig. 6.10 is due to the coarse discretisation used (15x15 collocation points), as evidenced by the accurate results presented in Fig. 6.11 where 25x25 collocation points were used.

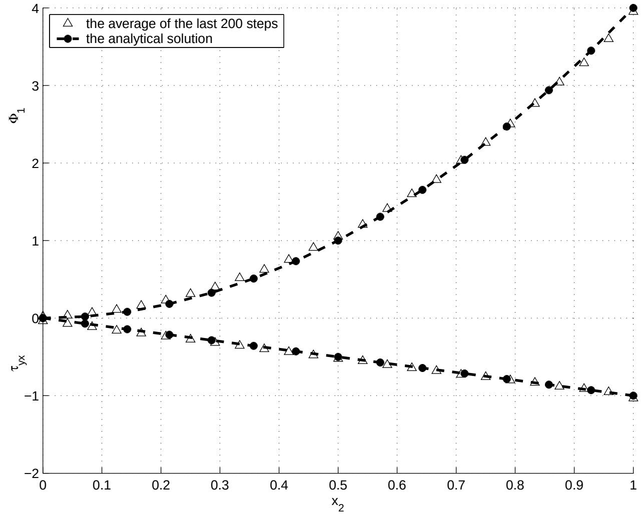


Figure 6.11: The steady state planar Poiseuille flow problem using the Hookean dumbbell (Oldroyd-B) model with 25×25 collocation points: the polymer shear stress and the first normal stress difference (averaged of the last 300 steps) on the middle plane $x_1 = 0.5$ with respect to x_2 are denoted by '$\triangle$'. The dashed line represents the analytical solution.

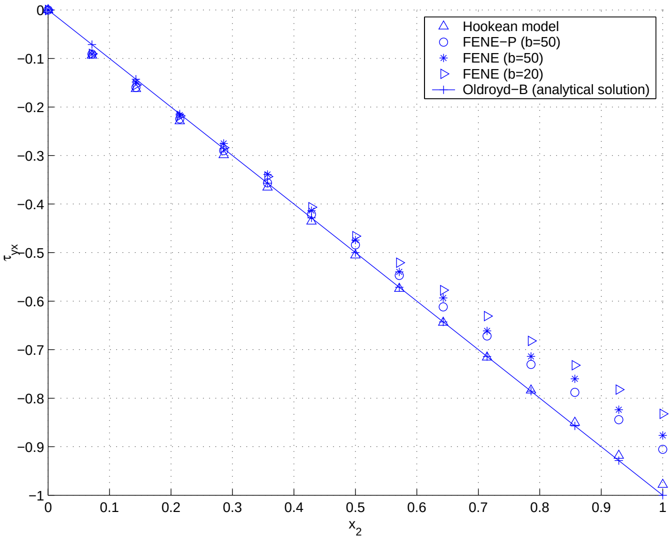


Figure 6.12: The steady state planar Poiseuille flow problem: comparison of the polymer shear stress on the middle plane $x_1 = 0.5$, '$\triangle$' for Hookean model, '$*$' for FENE (b=50), '$\diamond$' for FENE (b=20) using the present method and solid line through '+' for Oldroyd-B (Hookean) model using analytical method.

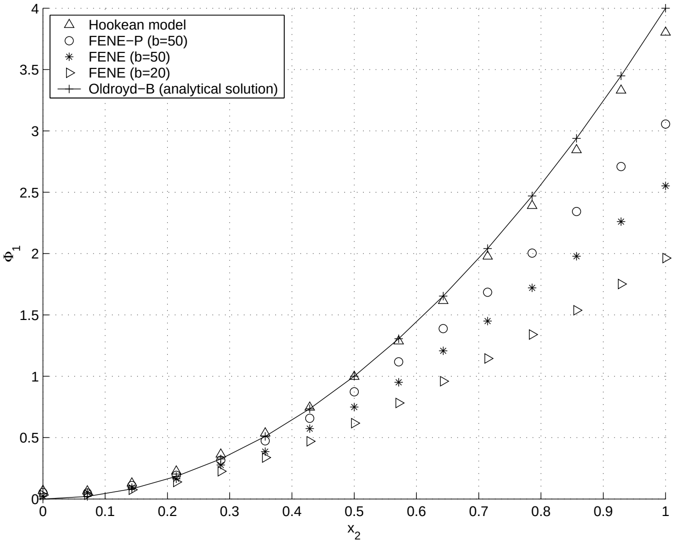


Figure 6.13: The steady state planar Poiseuille flow problem: comparison of the first normal stress difference on the middle plan $x_1 = 0.5$, '$\triangle$' for Hookean model, '$*$' for FENE (b=50), '$\diamond$' for FENE (b=20) using the present method and solid line through '+' for Oldroyd-B (Hookean) model using analytical method.

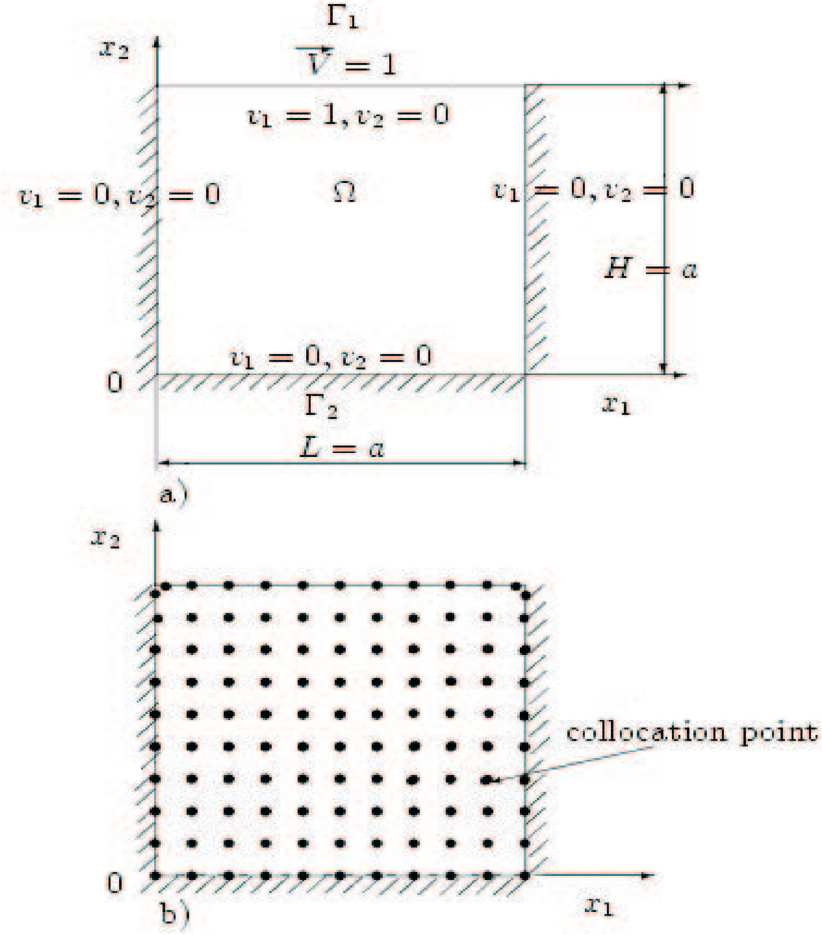


Figure 6.14: a) The lid driven square cavity problem: velocity of the upper lid is unity; non-slip boundary conditions apply at the fluid-solid interfaces. b) The collocation point distribution is only schematic.

6.5.3 Lid driven square cavity

While this problem has attracted the interest of many researchers in the case of viscous fluids, there are very few numerical results for viscoelastic fluids. Mendelson *et al.* (1982) and Grillet *et al.* (1999) use the FEM for the analysis and Tran-Cong *et al.* (2002) employs a BEM and RBF approach for the numerical solution for the Oldroyd-B model. On the other hand, Pakdel *et al.* (1997) performed experiments on an ideal Boger fluid. The results cited above are used here for qualitative comparison with the present results since the fluids used in those studies are different, except for the case of Tran-Cong *et al.* (2002), from the Hookean dumbbell model (Oldroyd-B model) used here. The flow is creeping, isothermal and in a steady state.

The geometry of the computational domain with the chosen coordinate system is shown in Fig. 6.14a. Let L and H be the width and height of cavity, respectively. Using six different sets of collocation points $(11\times 11+2)$; $(15\times 15+2)$; $(17\times 17+2)$, $(19\times 19+2)$, $(21\times 21+2)$ and $(41\times 41+2)$ whose schematic distribution is described in Fig. 6.14b, 1000 dumbbells are assigned at each collocation point.

The fluid parameters are given by

$$\alpha = \frac{\eta_N}{\eta_o} = 1/9; \quad \lambda_H = 1. \tag{6.35}$$

Let V be the speed of the lid. The Deborah numbers is given by

$$De = \frac{\lambda_H V}{H}.$$

Similar to the works of Tran-Cong *et al.* (2002) and Grillet *et al.* (1999), the Dirichlet boundary conditions are given, in dimensionless form, by (Fig. 6.14.a):

$$\begin{cases} u_1(\mathbf{x}) = 1 & \forall \mathbf{x} \in \Gamma_1, \\ u_2(\mathbf{x}) = 0 & \forall \mathbf{x} \in \Gamma_1, \\ \mathbf{u}(\mathbf{x}) = 0 & \forall \mathbf{x} \in \Gamma_2. \end{cases}$$

In order to reduce the number of the degrees of freedom of the problem, the penalty function method for the momentum equation (6.1) is employed and then the sum square error (6.16) is rewritten as follows:

$$\begin{aligned} SSE = & \sum_{\mathbf{x}_i\in\Omega}\left(\alpha\left[\frac{\partial^2 u_1}{\partial x_1^2}+\frac{\partial^2 u_1}{\partial x_2^2}\right]_{\mathbf{x}_i}+\frac{p_e}{\eta_o}\left[\frac{\partial^2 u_1}{\partial x_1^2}+\frac{\partial^2 u_1}{\partial x_1\partial x_2}\right]_{\mathbf{x}_i}+\phi_{1,i}\right)^2 + \\ & \sum_{\mathbf{x}_i\in\Omega}\left(\alpha\left[\frac{\partial^2 u_2}{\partial x_1^2}+\frac{\partial^2 u_2}{\partial x_2^2}\right]_{\mathbf{x}_i}+\frac{p_e}{\eta_o}\left[\frac{\partial^2 u_2}{\partial x_2^2}+\frac{\partial^2 u_2}{\partial x_1\partial x_2}\right]_{\mathbf{x}_i}+\phi_{2,i}\right)^2 + \\ & \sum_{\mathbf{x}_i\in\Gamma_1}(u_1-1)^2_{\mathbf{x}_i}+\sum_{\mathbf{x}_i\in\Gamma_1}(u_2)^2_{\mathbf{x}_i}+\sum_{\mathbf{x}_i\in\Gamma_2}(u_1)^2_{\mathbf{x}_i}+\sum_{\mathbf{x}_i\in\Gamma_2}(u_2)^2_{\mathbf{x}_i}. \end{aligned} \tag{6.36}$$

where $\phi_{1,i} = \frac{\partial \tau_{11}^p}{\partial x_1}(\mathbf{x}_i) + \frac{\partial \tau_{21}^p}{\partial x_2}(\mathbf{x}_i)$ and $\phi_{2,i} = \frac{\partial \tau_{12}^p}{\partial x_1}(\mathbf{x}_i) + \frac{\partial \tau_{22}^p}{\partial x_2}(\mathbf{x}_i)$ are known by virtue of the BCFs simulation and approximated using TPS-RBFNs and $(.)_{\mathbf{x}_i}$ denotes the value of $(.)$ at $\mathbf{x}_i$.

For the discretisation with $(41 \times 41 + 2)$ collocation points, the iterative non-overlapping domain decomposition technique is employed in order to deal with the dense matrix and its possible ill-conditioning. The result obtained as depicted in Fig. 6.16, is carried out using four equal sub-domains. The domain decomposition problem is detailed in the next chapter.

Results and discussion

In order to demonstrate that numerical solutions converge to the correct solution, six different sets of collocation points are used as described above and 'mesh convergence' is measured by the following criterion

$$CR = \sqrt{\frac{\sum\limits_{tp}\sum\limits_{i=1}^{2}\left(u_i^n - u_i^{n-1}\right)^2}{\sum\limits_{tp}\sum\limits_{i=1}^{2}\left(u_i^n\right)^2}}, \tag{6.37}$$

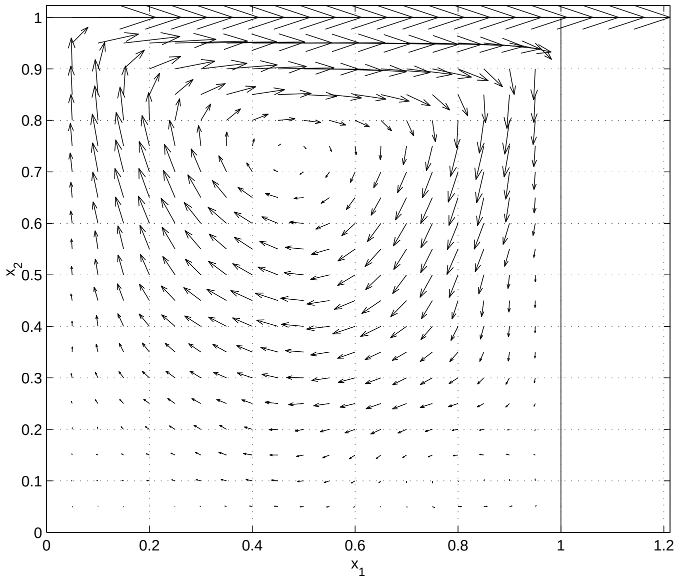
where tp is the set of internal test points, u_i^{n-1} is the ith component of the velocity at an internal test point associated with the coarser discretisation and u_i^n is the corresponding quantity associated with the finer one. Since the solution (velocity field) is the average of the results of a number of iterations, u_i^{n-1} and u_i^n are the average values at the internal test points. Table 6.1 reports the trend of CR for the velocity field with increasing collocation density for $De = 1.0$. The process is deemed to have achieved 'mesh convergence' when CR is $O(10^{-2})$. As in the previous examples, the numerical solutions are the average of the results of the last 200 iterations after convergence. In the case of the Hookean dumbbell model, the result is in good agreement with the findings of Tran-Cong *et al.* (2002). Figs. 6.15 (collocation density $21 \times 21 + 2$) and 6.16 (collocation density $41 \times 41 + 2$) depict the velocity field for $De = 1$ and Figs. 6.17 describes

Table 6.1: The lid driven square cavity flow problem using the Hookean dumbbell model: Trend of the 'mesh convergence' measure, CR defined by (6.37) with increasing number of collocation points for $De = 1$. N: number of collocation points, tp: number of internal test points.

N	tp	CR
$11 \times 11 + 2$	81	1.0000
$15 \times 15 + 2$	81	0.0447
$17 \times 17 + 2$	169	0.0123
$19 \times 19 + 2$	225	0.0116
$21 \times 21 + 2$	289	0.0097
$41 \times 41 + 2$	443	0.0093

the x_1-component velocity profile on the vertical central plane $x_1 = 0.5$ and the x_2-component velocity profile on the horizontal central plane $x_2 = 0.5$.

Fig. 6.18 depicts the velocity field for $De = 1.5$. The results show that the primary vortex center tends to shift upstream and towards the driving lid when De increases. The primary vortex appears to extend up to the walls as shown in Fig. 6.16 where the size and location of secondary vortices can also be observed in the lower left and right corners. Although the present results can only be compared with Tran-Cong *et al.* (2002) as they used the same model fluid (Oldroyd-B), it is generally in qualitative agreement on the typical flow features reported by Grillet *et al.* (1999), Mendelson *et al.* (1982), and Pakdel *et al.* (1997) in which the vortex shifts upstream as the De number increases. Since the discussion on the efficiency of the control variate variance reduction is not the object of the present work, only an observation of the effect of the number of configuration fields on the velocity fields is given in Fig. 6.19 for the cases of 600, 1000 and 1400 dumbbells assigned at each collocation point and $De = 1$. The results shown in Fig. 6.19 demonstrate that the choice of 1000 dumbbells is adequate.



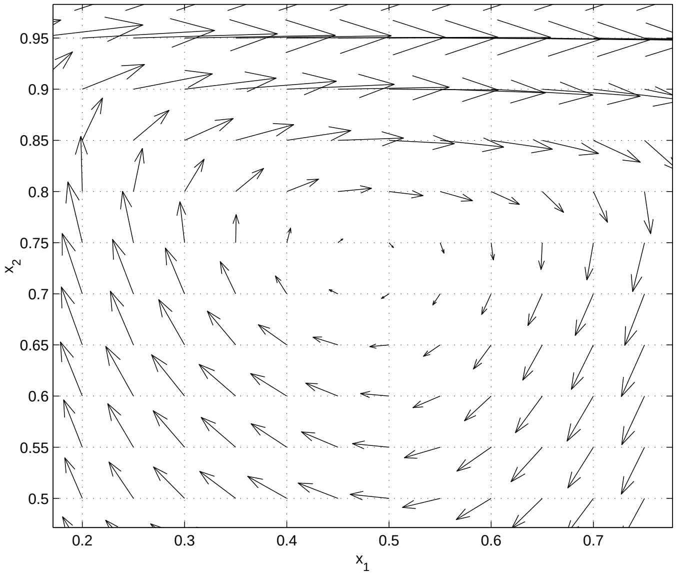


Figure 6.15: The lid driven square cavity flow problem using the Hookean dumbbell model: The velocity field (upper figure). The zoomed velocity field around the vortex position (lower figure). The parameters are $\alpha = 1/9$, $\lambda_H = 1 \quad (De = 1)$, $(21 \times 21 + 2)$ collocation points.

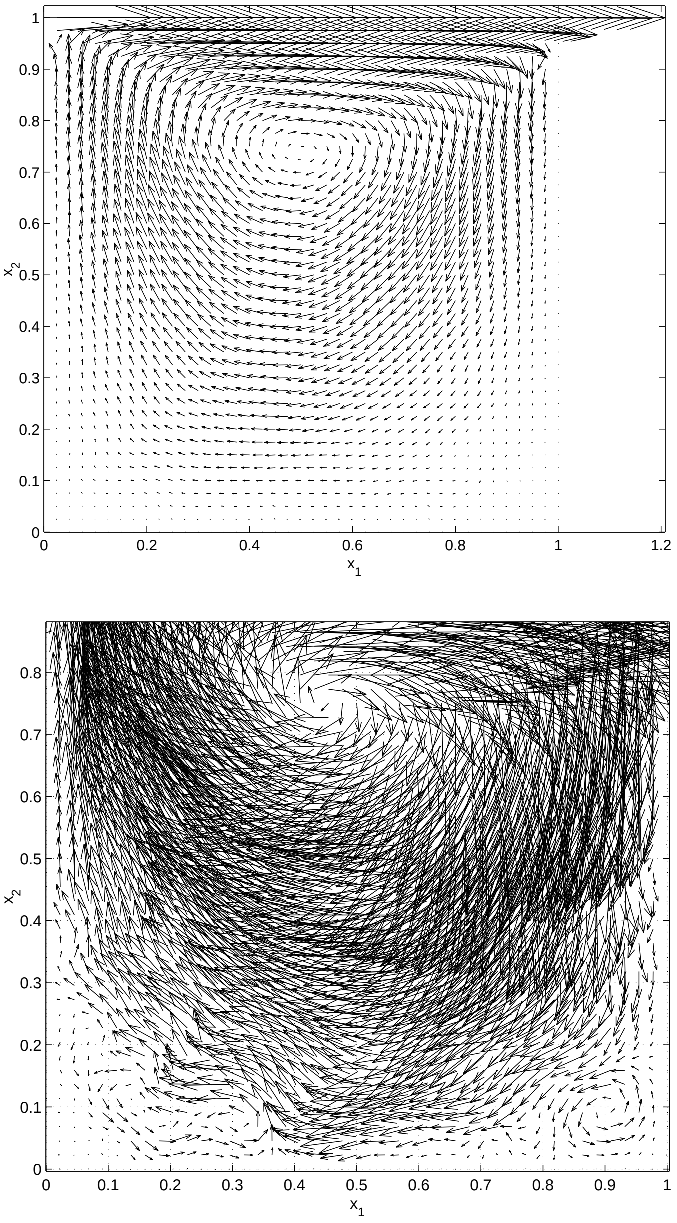


Figure 6.16: The lid driven square cavity flow problem using the Hookean dumbbell model: the velocity field (upper figure); the zoomed velocity field around the vortex positions. The parameters are $\alpha = 1/9$, $\lambda_H = 1 \quad (De = 1)$, $(41 \times 41 + 2)$ collocation points.

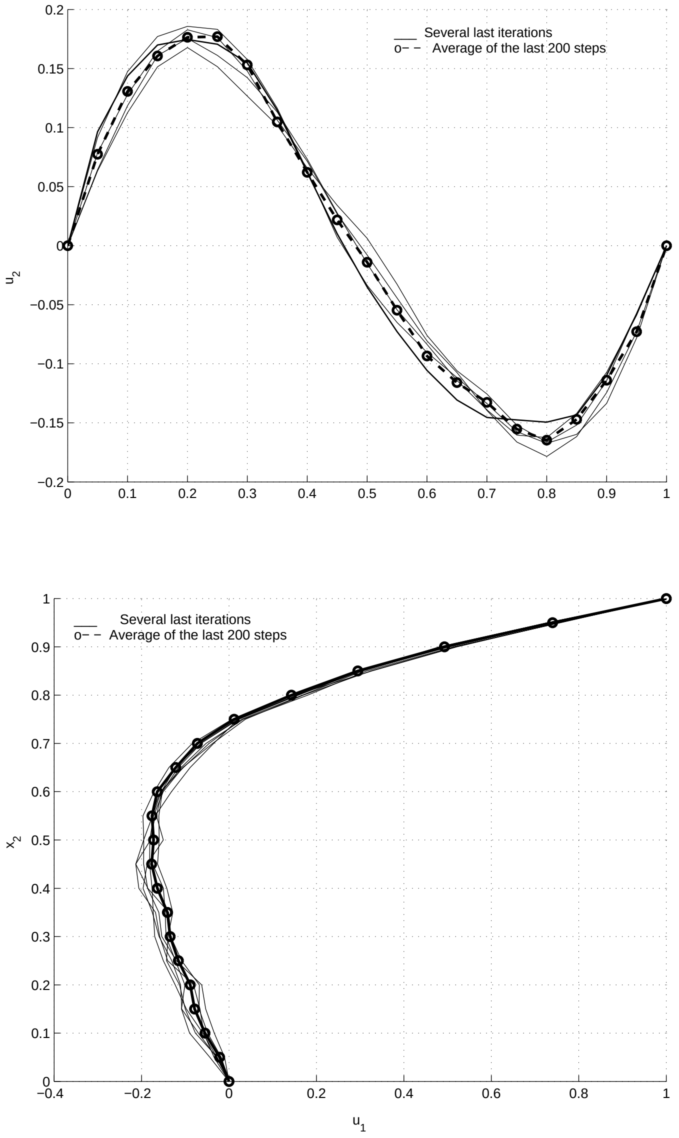


Figure 6.17: The lid driven square cavity flow problem using the Hookean dumbbell model: the profile of the velocity component u_2 on the horizontal central plane (upper figure). The profile of the velocity component u_1 on the vertical central plane (lower figure). The solid lines are for the last several steps and $-o-$ denotes the average of the results from the last 200 iterations. The parameters are the same as in Fig. 6.15.

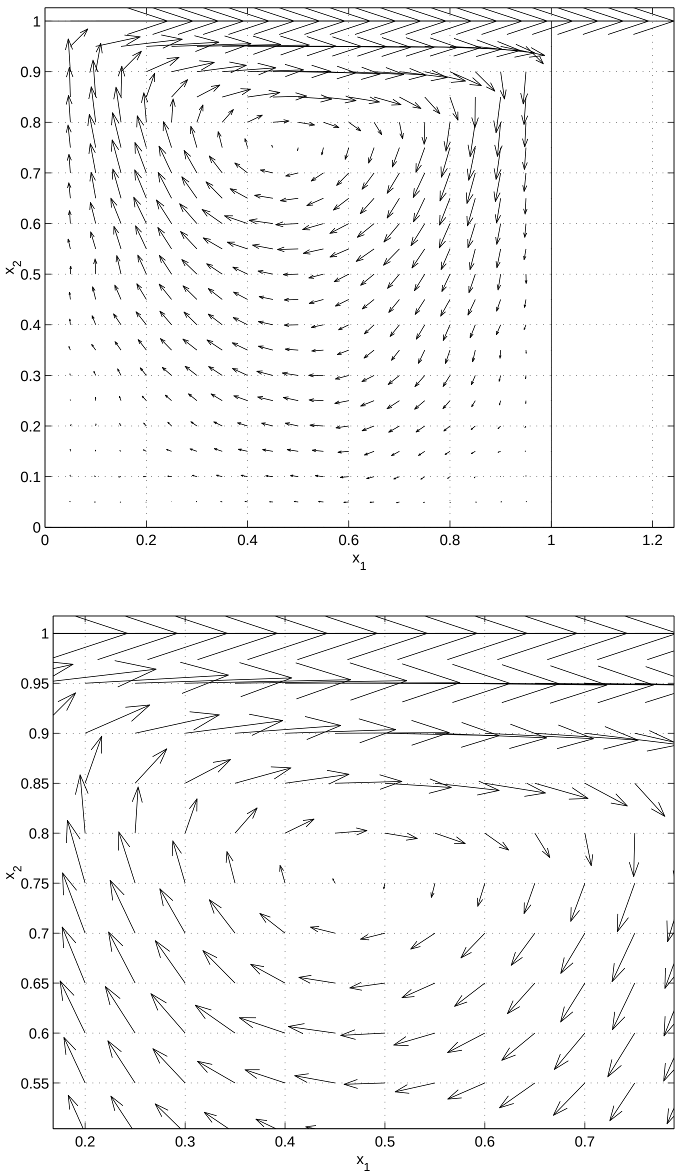


Figure 6.18: The lid driven square cavity flow problem using the Hookean dumbbell model: The velocity field (upper figure). The zoomed velocity field around the vortex position (lower figure). The parameters are same as in Fig. 6.15 except that $\lambda_H = 1.5 \quad (De = 1.5)$.

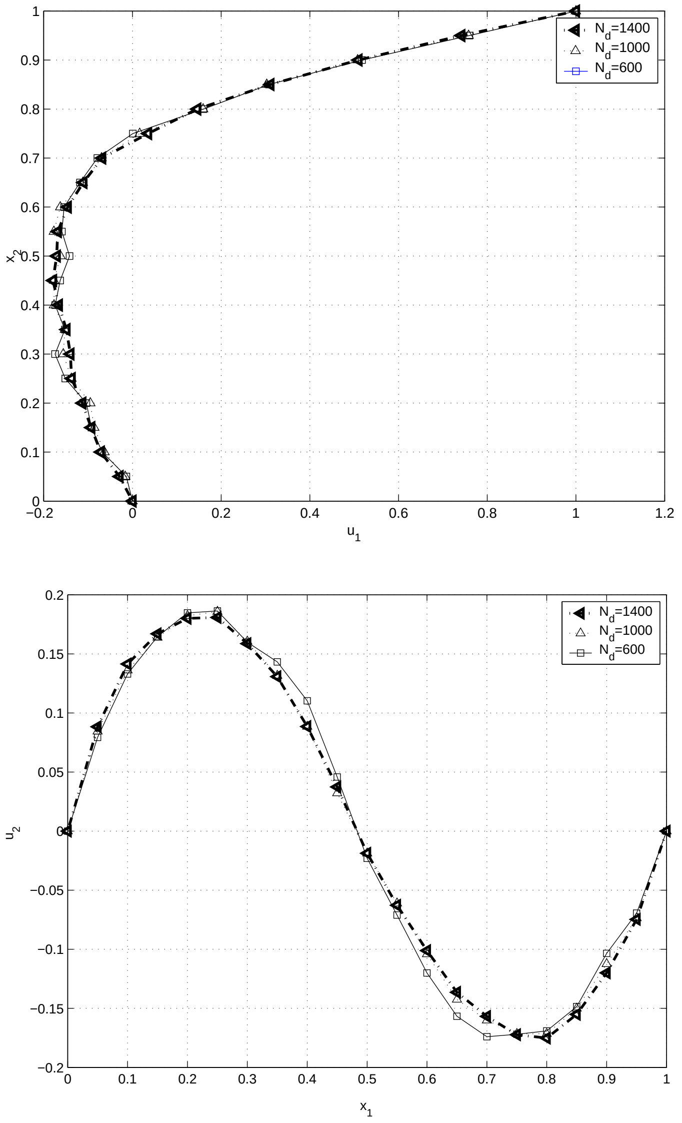


Figure 6.19: The lid driven square cavity flow problem using the Hookean dumbbell model: the profile of the velocity component u_2 on the horizontal central plane (lower figure). The profile of the velocity component u_1 on the vertical central plane (upper figure). The average of the results from the last 200 iterations corresponding to the number of dumbbells assigned at each collocation point 600, 1000 and 1400. The parameters are the same as in Fig. 6.15, De = 1.

6.6 Concluding remarks

This chapter reports the development of a computational method for viscoelastic flows using a combination of a RBFN-based element-free method and SST from the Eulerian point of view for 1D and 2D problems.

The main advantages of the present method are that: particle tracking is not necessary; variance reduction of the stochastic stress tensor is achieved for the same number of dumbbells used; the noise effect due to the Brownian component is reduced; the method is element-free in both macroscopic and microscopic parts and only an unstructured set of collocation points is required to discretise all governing equations.

The method is verified with standard test problems, namely the start up Couette flow and the planar Poiseuille flow problems. The potential of the method is demonstrated with the successful solution of a non trivial problem, namely the lid-driven square cavity problem. Besides the advantages, the drawback of the present method when solving large scale problems is the possibility that coefficient matrices are ill conditioned. This disadvantage can be overcome by means of a domain decomposition technique as described in the next chapter.

Chapter 7

Parallel mesh-free TPS-RBFN-BCF with Domain Decomposition for dilute polymer solution

This chapter reports the suitability of a domain decomposition technique for the hybrid simulation of dilute polymer solution flows using Eulerian Brownian dynamics and RBFN-based methods (Tran-Canh and Tran-Cong, 2003c). As shown in the previous chapter, the RBFN-BCF method incorporates the features of the BCFs scheme (which render both closed form constitutive equations and a particle tracking process unnecessary) and a meshless method (which eliminates element-based discretisation of domains). However, when dealing with large scale problems, there appear several difficulties: the high computational time associated with the SST, and the ill-condition of the system matrix associated with the RBFNs. One way to overcome these disadvantages is to use parallel domain decomposition (DD) techniques. This approach makes the BCF-RBFN method more suitable for large scale problems

7.1 Introduction

In this chapter, the RBFN-BCF method is considered in conjunction with domain decomposition techniques for approximation of functions and solution of PDEs. The domain under consideration is partitioned into a number of subdomains and the task on each subdomain is carried out on a distributed memory computer with a parallelization procedure. Furthermore, since the tasks of the stochastic and deterministic processes such as solving the SDEs for dumbbells and the PDEs governing the flow, can be carried out separately in subdomains, the parallel DD techniques can be applied effectively in both macroscopic and microscopic components. In general either only one subdomain or a group of sub-domains can be handled by a separate processor, the former case is considered here. Hence, the parallel domain decomposition method increases the throughput, and, at the same time, removes the problem of ill-conditioning of the system matrix associated with the RBFN-based method (discrete ill-posed problem). For the purpose of function approximation, the domain decomposition is relative simple in the sense that the data on the interfaces between the sub-domains are known. It is more complex for solving PDEs in applying the DD method. Since the interfaces between the subregions are considered as part of the boundary of the sub-domains, the associated unknowns need to be determined as part of the solution of the problem.

The domain decomposition techniques can be classified into two main categories, namely the sub-regioning and the substructuring DD methods. The former includes iterative non-overlapping schemes (Marini and Quarteroni, 1988; Funaro *et al.*, 1988; Yang, 1996), and the iterative overlapping schemes using the additive or multiplicative Schwarz techniques (Zhou *et al.*, 2003). While the subregioning methods focus on the way to estimate the boundary conditions on the interfaces that ensures the continuity and the smoothness across the interfaces, the substructuring DD method is based on explicit computation and factorization of a sequence of Schur complement matrices. These techniques circumvent the ill-posed problem resulted from using the RBFs as a global interpolant, reduce the memory requirement and achieve a high accuracy of the solution. In the present work, the iterative non-overlapping domain decomposition method is employed in conjunction with the hybrid RBFN-BCF simulation. The chapter is organized as follows. In §7.2, the RBFN-BCF method for dilute polymer solution is reviewed briefly. §7.3 is to present the domain decomposition for the RBFN-based method in which the iterative non-overlapping DD for solving PDEs is described. The parallelization of the BCF-RBFN method is reported in §7.4. Numerical examples are then discussed in §7.5 in which the steady state 10:1 planar contraction and lid driven cavity flows are simulated, followed by a brief conclusion in §7.6.

7.2 Review of TPS-RBFN-BCF method for dilute polymer solutions

A dilute polymer solution is considered as an incompressible suspension of non-interacting dumbbells in a Newtonian solvent. Dumbbells are characterized by connector vectors **Q**'s. These connector vectors, which constitute Brownian Configuration Fields (Hulsen *et al.*, 1997), satisfy the dimensionless form of the SDE as follows

$$d\mathbf{Q} = \left[-\mathrm{De} \left(\mathbf{u} \cdot \boldsymbol{\nabla} \mathbf{Q} + \boldsymbol{\kappa} \cdot \mathbf{Q} \right) - \frac{1}{2} \mathbf{F}(\mathbf{Q}) \right] dt + d\mathbf{W}(t), \tag{7.1}$$

where $\mathbf{Q}$, $\mathbf{u}$, t are the dimensionless forms of connector vectors, velocity fields and time, scaled by $(k_B T/H)^{1/2}$, V, λ_H, respectively; $De = \lambda_H/\lambda_{flow}$ is the Deborah number; $\lambda_H = \varsigma/4H$ and $\lambda_{flow} = a/V$ are characteristic relaxation times of the fluid and flow, respectively; ς is the friction coefficient between the dumbbell and the solvent; H is the spring constant; a, V are characteristic length and velocity; k_B is the Boltzmann constant; T is the absolute temperature; $\boldsymbol{\kappa}$ is the velocity gradient; $\mathbf{W}(t)$ is a Wiener process with mean $\langle W_i(t) \rangle = 0$

and covariance $\langle W_i(t)W_j(t')\rangle = \delta_{ij}\min(t,t')$ and accounts for the random displacement of the beads due to thermal motion, and $\mathbf{F}$ is the dimensionless spring force which is given by, for the FENE and FENE-P models, respectively

$$\mathbf{F}_{\text{FENE}} = \frac{\mathbf{Q}}{1-(\mathbf{Q}^2/b)}, \quad \mathbf{F}_{\text{FENE-P}} = \frac{\mathbf{Q}}{1-\langle\mathbf{Q}^2\rangle/b}, \tag{7.2}$$

where $b = HQ_o^2/k_BT$ is the square of the dimensionless maximum extensibility. Based on the Brownian configuration fields, the polymer stress tensor is determined by Kramer formula as

$$\boldsymbol{\tau}^p = -\langle\mathbf{Q}.\mathbf{F}(\mathbf{Q})\rangle + \mathbf{I}, \tag{7.3}$$

where $\boldsymbol{\tau}^p$ is the dimensionless form of the polymer stress tensor scaled by n_dk_BT; n_d is the density of dumbbells; $\mathbf{I}$ is the unity matrix. The velocity field in Eq (7.1) of steady, isothermal, incompressible flows is governed by the equation of momentum (using the penalty function method) as follows

$$2\alpha\boldsymbol{\nabla}\cdot\mathbf{D} - Re(\mathbf{u}\cdot\nabla)\mathbf{u} + \beta_1\boldsymbol{\nabla}\left(\boldsymbol{\nabla}\cdot\mathbf{u}\right) \;=\; -\beta_2\nabla\cdot\boldsymbol{\tau}^p, \tag{7.4}$$

where $p = -p_e(\boldsymbol{\nabla}\cdot\mathbf{u})$; p_e is a sufficiently large parameter (see §5.2); $\alpha = \eta_N/\eta_o$; $Re = \rho Va/\eta_o$; $\eta_o = \eta_N + \eta_P$; η_N, η_P are the solvent and polymer viscosities; ρ is the fluid density; $\beta_1 = p_e/\eta_o$; $\beta_2 = n_dk_BT/(\eta_oV/a)$ is a dimensionless parameter; p is the pressure arisen from the incompressible constraint; $\mathbf{D}$ and $\boldsymbol{\tau}_p$ denote the dimensionless form of the rate of strain tensor and polymer stress, respectively. While $\boldsymbol{\tau}^p$ is determined by the Brownian configuration field method, the solutions ($\mathbf{u}$) of the governing PDEs (Eq (7.4)) are obtained by using the TPS-RBFN-based mesh-free method described in the previous chapter.

7.3 Domain decomposition method for RBFN-based element-free method

Although the parallel domain decomposition schemes were introduced a long time ago with the advent of powerful supercomputers, they have been well developed only for the element type methods (Quarteroni and Valli, 1999 and Smith *et al.*, 1996), and there are only a few attempts to couple RBFN based meshless methods and DD techniques (Dubal, 1994; Beatson *et al.*, 2000 and Zhou and Hon, 2003).

The non-overlapping domain decompositions are easy for parallel implementation and they also have some advantages over the overlapping method, for example they are efficient for handling elliptic problems with large jumps in coefficients (Xu and Zou, 1998). For further references, these techniques can be found in, for example, Smith *et al.* (1996). A very good survey on the non-overlapping DD method was given by Xu and Zou (1998). In this work, the iterative non-overlapping domain decomposition is employed to couple with RBFN-based mesh-free method for the function approximation and numerical solution of PDEs. The method is obviously well suited for parallel computing architectures. The implementation is for both macroscopic and microscopic parts of the method. For the purpose of function approximation, since the data are known on the interfaces, the DD techniques is very simple and not presented here. For solving PDEs, the interfaces are considered as part of the boundary of sub-domains and the associated boundary conditions need to be determined.

7.3.1 Review of TPS-RBFN-based method

Consider a general steady-state problem as follows

$$\mathcal{L}u(\mathbf{x}) = y \quad \forall x \in \Omega, \tag{7.5}$$

$$\mathcal{B}u(\mathbf{x}) = b \quad \forall x \in \partial\Omega, \tag{7.6}$$

where Ω is the volume under consideration; $\partial\Omega$ is the boundary of Ω; $\mathcal{L}$ is an arbitrary differential operator; $\mathcal{B}$ is an operator imposed as boundary conditions, such as Dirichlet, Newmann or a mixture of both; u is an unknown function; y

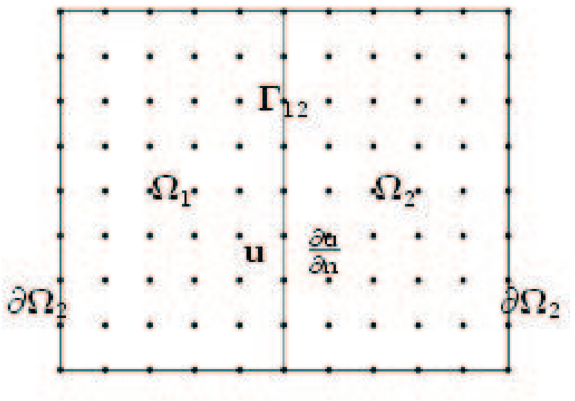


Figure 7.1: The schematic overlapping domain decomposition techniques with two and four sub-domains

and b are given functions. The solution u can be expressed as follows

$$u \approx \sum_{j=1}^{m} w^j h^j(\mathbf{x}) + \sum_{k=1}^{\bar{m}} \iota^k p^k(\mathbf{x}) \tag{7.7}$$

where w^j's and ι^k are the weights; h^j is the chosen radial basis function corresponding to the j^{th} RBF-neuron; p^j is the polynomial basis function of the k^{th} PBF. The partial derivatives of $f(\mathbf{x})$ can be calculated analytically as follows

$$\mathcal{L}u(\mathbf{x}) \approx \sum_{j=1}^{m} w^j \mathcal{L}h^j(\mathbf{x}) + \sum_{k=1}^{\bar{m}} \iota^k \mathcal{L}p^k(\mathbf{x}), \tag{7.8}$$

where $\mathcal{L}$ is a derivative operator. The RBFs h^j employed here are TPS-RBF, the chosen PBF is 2-D linear; $\mathcal{L}h^j$ for TPS-RBFN is given in §5.5. The substitution (7.7) and (7.8) into (7.5) and (7.6) followed by the application of the general linear least squares principle lead to a system of equations in the unknowns w^j's and ι^k's, which was described in §5.5.2.

7.3.2 The iterative non-overlapping DD scheme for RBFN

For illustration, reconsidering a steady state problem governed by (7.5) and (7.6) using DD technique with two sub-domains (see Fig. 7.1). The domain under consideration is divided into 2 non-overlapping sub-domains Ω_1, Ω_2. Let

$\partial\Omega_1$ and $\partial\Omega_2$ be the boundaries of Ω_1 and Ω_2, respectively; Γ_{12} is the artificial boundary (interface) between Ω_1 and Ω_2. The boundary condition imposed on this interface can be Dirichlet-Neumann, Neumann-Neumann or otherwise. In this chapter, the boundary condition imposed on the interface Γ_{12} is of Dirichlet-Neumann type. The algorithm of the iterative non-overlapping DD consists of two main tasks. (i) Determining the unknowns on interfaces based on the compatibility conditions on the interfaces. (ii) Solving the smaller boundary value problems on the sub-domains. This algorithm is written for the subdomains Ω_1 and Ω_2, at the step $i+1$, respectively, as follows

$$\begin{aligned} \mathcal{L}u_1^{i+1}(\mathbf{x}) &= y \quad \mathbf{x} \in \Omega_1, \\ \mathcal{B}u_1^{i+1}(\mathbf{x}) &= b \quad \mathbf{x} \in \partial\Omega_1 \setminus \Gamma_{12}, \\ f_1^{i+1}(\mathbf{x}) &= c^i \quad \mathbf{x} \in \Gamma_{12}; \end{aligned} \tag{7.9}$$

and

$$\begin{aligned} \mathcal{L}u_2^{i+1}(\mathbf{x}) &= y \quad \mathbf{x} \in \Omega_2, \\ \mathcal{B}u_2^{i+1}(\mathbf{x}) &= b \quad \mathbf{x} \in \partial\Omega_2 \setminus \Gamma_{12}, \\ \frac{\partial u_2^{i+1}}{\partial \mathbf{n}}(\mathbf{x}) &= \frac{\partial u_1^{i}}{\partial \mathbf{n}}(\mathbf{x}) \quad \mathbf{x} \in \Gamma_{12}, \end{aligned} \tag{7.10}$$

where c^i is extracted from Ω_2 (the neighboring sub-domain sharing the interface with Ω_1) at the step i and given by

$$c^i = \theta c^{i-1} + (1-\theta)u_2^i; \quad 0 \le \theta \le 1,$$

u_k^i $(k = 1, 2)$ is the numerical solutions obtained from the subdomain Ω_k at the collocation points located on the interface at a step i and $\frac{\partial u}{\partial \mathbf{n}} = \frac{\partial u}{\partial n_1} n_1 + \frac{\partial u}{\partial n_2} n_2$. This algorithm can be extended for $N > 2$ subdomains by using the black and white coloring method (Quarteroni, 1999). In order to facilitate the parallel implementation, the block-parallel coloring scheme is used in this work. Let Ω_k be a subdomain $(1 \le k \le N)$, $\Gamma_{kl} = \Omega_k \cap \Omega_l$ the common interface between two adjoining subdomain Ω_k and Ω_l, G_b and G_w be the two groups of sub-domains

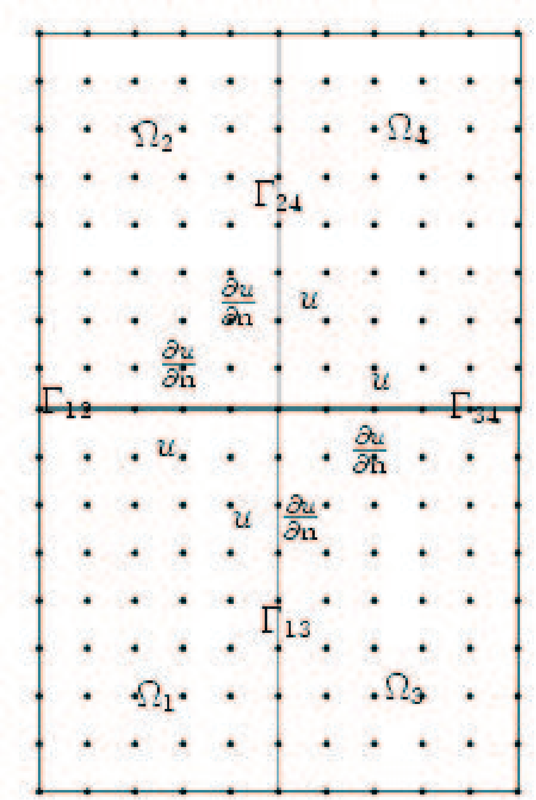

Ω_2
Ω_4
Γ_{24}
$\frac{\partial u}{\partial n}$
u
$\frac{\partial u}{\partial n}$
u
Γ_{12}
Γ_{34}
u
$\frac{\partial u}{\partial n}$
u
$\frac{\partial u}{\partial n}$
Γ_{13}
Ω_1
Ω_3

Figure 7.2: The iterative non-overlapping domain decomposition techniques using black white coloring method:$N = 4$ sub-domains; $G_b \equiv [\Omega_1, \Omega_4]$ and $G_w \equiv [\Omega_2, \Omega_3]$.

which are defined by (Fig. 7.2)

$$\begin{aligned} G_b &= \{\boldsymbol{\Omega}_k \text{ is black }, 1 \leq k \leq N\}, \\ G_w &= \{\boldsymbol{\Omega}_k \text{ is white }, 1 \leq k \leq N\}. \end{aligned}$$

The algorithm of this method is given by

$$\begin{cases} Lu_k^{i+1}(\mathbf{x}) = y \quad x \in \Omega_k \quad k \in G_b \\ u_k^{i+1}(\mathbf{x}) = \theta u_l^i + (1-\theta)\, u_k^i \quad \mathbf{x} \in \Gamma_{kl}, l \in G_w, \Gamma_{kl} \neq 0 \end{cases} \tag{7.11}$$

and

$$\begin{cases} Lu_l^{i+1}(\mathbf{x}) = y \quad \mathbf{x} \in \Omega_l \quad l \in G_w \\ \frac{\partial u_l^{i+1}}{\partial \mathbf{n}}(x) = \frac{\partial u_l^i}{\partial \mathbf{n}}(\mathbf{x}) \quad \mathbf{x} \in \Gamma_{kl}, k \in G_b, \Gamma_{kl} \neq 0 \end{cases} \tag{7.12}$$

Hence, the sub-programs corresponding to subdomains Ω_k are independent of one another and they are solved simultaneously with separate CPUs. If any part of $\partial\Omega_k$ coincides with the original boundary $\partial\Omega$, the original boundary conditions would apply there as appropriate.

The overall procedure can now be described as follows

(1.) Divide the analysis domain into a number of subdomains and determine G_b and G_w and the interfaces between subdomains.

(2.) Guess initial boundary conditions at the interfaces for subdomains as appropriate (Dirichlet condition for $\Omega_k \in G_b$ and Neumann condition for $\Omega_k \in G_w$).

(3.) Solve the boundary value problems described by (7.9)-(7.10) corresponding to Ω_k, $(1 \leq k \leq N)$ using TPS-RBFN-based element-free method on separate CPUs

(4.) Check for the compatibility on the whole artificial interfaces.

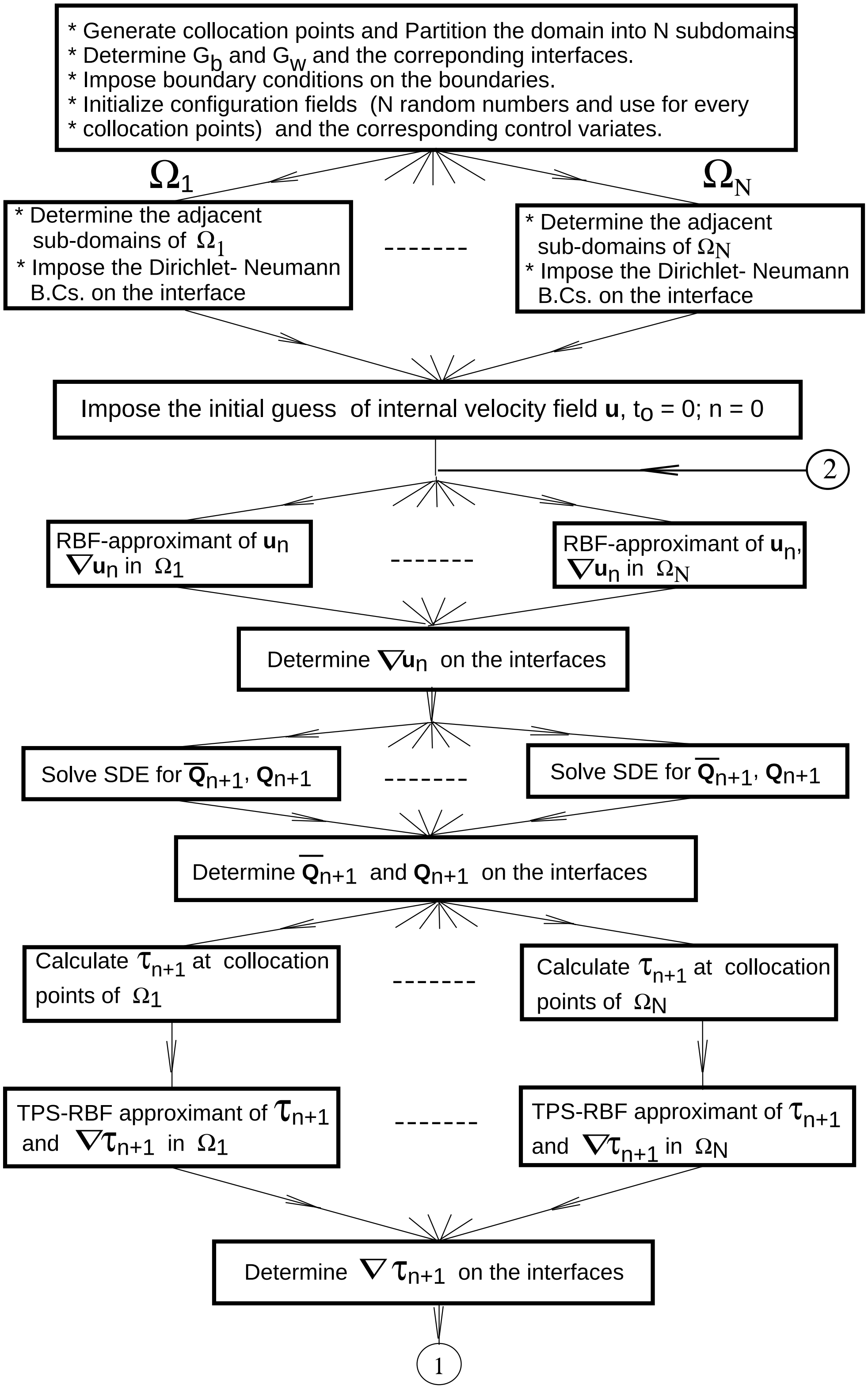
(5.) If the procedure is not yet converged, return step 3, when the boundary conditions on all interfaces are updated as described in the second equation of (7.11) for $\Omega_k \in G_b$ and (7.12) for $\Omega_l \in G_w$.

(6.) stop the procedure.

The singular value decomposition scheme is employed to get the reciprocal matrices associated with each subregion.

7.4 Parallelization and DD scheme for RBFN-BCF method

The parallelization performed in the present work is focussed on both micro and macro parts as mentioned in the previous section. The parallelization is carried out in regard to collocation points and based on the domain decomposition technique. In this work the non-overlapping scheme is employed. The basic parallelization scheme is described in the Figs. 7.3 and 7.4. Parallel implementation of the algorithm is based on the single program, multiple data (SPMD) paradigm with message passing interface (MPI) for parallel communication. Although a CPU could accommodate more than one sub-domain, in the present work, only one sub-domain is allowed per processor. The parallel implementation is carried out on the AlphaServer ES45 installed at the Australian Partnership for Advanced Computing National Facility. It features 127 nodes, each contains $4 \times 1GHz$ ev68 cpus with between 4 and $16GB$ of RAM (total of 700GB of RAM), see *www.nf.apac.edu.au/facilities/sc/hardware.php* for more details. In this thesis, we use Fortran-90 (Press *et al.*, 1992) supported by the MPI (Gropp *et al.*, 1998; Snir *et al.*, 1998a,b) in parallel programming. MPI is a library specification for message-passing and designed for high performance on both massively parallel machines and on workstation clusters. Processor loads are almost balanced with static balancing where sub-domains contain the same number collocation points and then the same number of dumbbells. Hence, the parallelization for both microscopic and macroscopic levels is carried out on each subdomain.


* Generate collocation points and Partition the domain into N subdomains
* Determine G_b and G_w and the correponding interfaces.
* Impose boundary conditions on the boundaries.
* Initialize configuration fields (N random numbers and use for every
* collocation points) and the corresponding control variates.
Ω_1
Ω_N
* Determine the adjacent sub-domains of Ω_1
* Impose the Dirichlet- Neumann B.Cs. on the interface
----
* Determine the adjacent sub-domains of Ω_N
* Impose the Dirichlet- Neumann B.Cs. on the interface
Impose the initial guess of internal velocity field **u**, $t_o = 0$; n = 0
2
RBF-approximant of $\mathbf{u}_n$ $\nabla\mathbf{u}_n$ in Ω_1

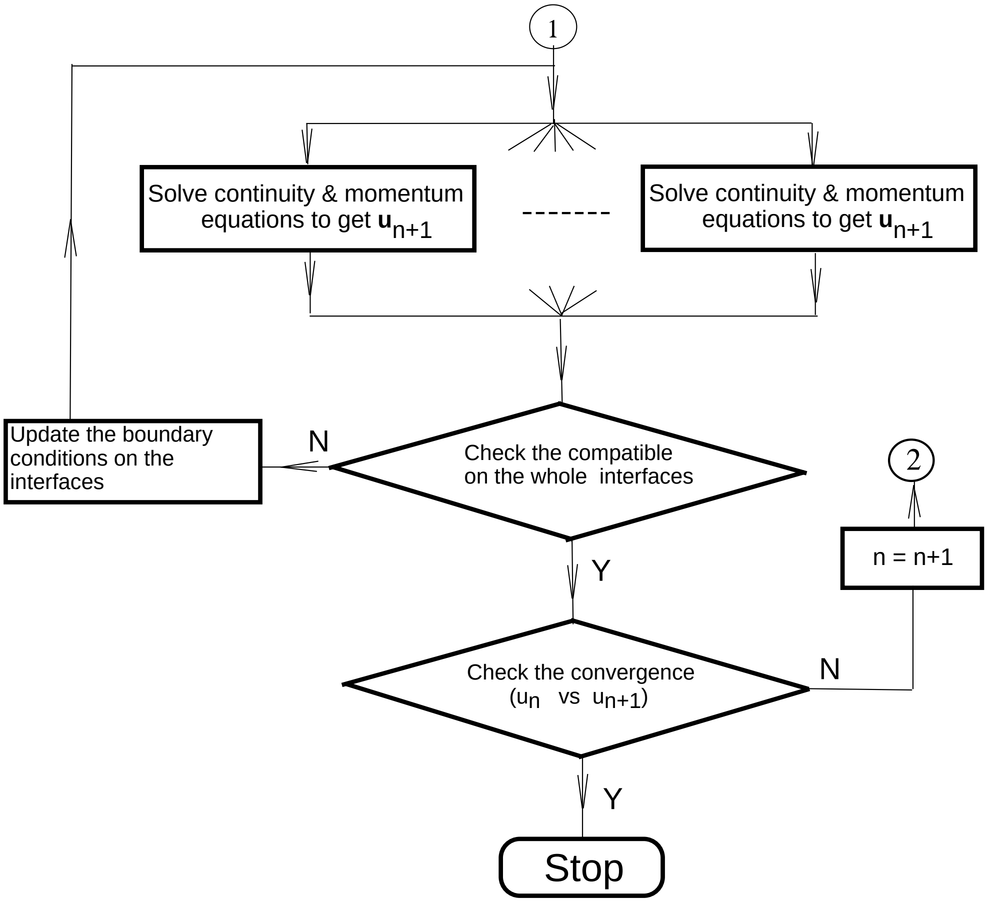
RBF-approximant of $\mathbf{u}_n$, $\nabla\mathbf{u}_n$ in Ω_N
Determine $\nabla\mathbf{u}_n$ on the interfaces
Solve SDE for $\overline{\mathbf{Q}}_{n+1}$, $\mathbf{Q}_{n+1}$

Solve SDE for $\overline{\mathbf{Q}}_{n+1}$, $\mathbf{Q}_{n+1}$
Determine $\overline{\mathbf{Q}}_{n+1}$ and $\mathbf{Q}_{n+1}$ on the interfaces
Calculate τ_{n+1} at collocation points of Ω_1

Calculate τ_{n+1} at collocation points of Ω_N
TPS-RBF approximant of τ_{n+1} and $\nabla\tau_{n+1}$ in Ω_1

TPS-RBF approximant of τ_{n+1} and $\nabla\tau_{n+1}$ in Ω_N
Determine $\nabla\tau_{n+1}$ on the interfaces
1

Figure 7.3: General flow-chart of the present RBFN-BCF-based element-free method in conjunction with the iterative non-overlapping domain decomposition technique. See next page.


1
Solve continuity & momentum equations to get $\mathbf{u}_{n+1}$

Solve continuity & momentum equations to get $\mathbf{u}_{n+1}$
Update the boundary conditions on the interfaces
N
Check the compatible on the whole interfaces
2
n = n+1
Y
Check the convergence (u_n vs u_{n+1})
N
Y
Stop

Figure 7.4: General flow-chart of the present RBFN-BCF-based element-free method in conjunction with the iterative non-overlapping domain decomposition technique. See previous page.

7.5 Numerical examples

In this section, the ability of the RBFN-BCF method in conjunction with parallel domain decomposition technique to simulate challenging problems is estimated, using the steady state planar 10:1 contraction and lid driven cavity flows of several single dumbbell model fluids. As already discussed in Chapter 6, if different random numbers are generated for the initial configuration, different trajectories of velocity and stresses will be experienced but the final average results do not change, which is a characteristic of stochastic simulations (Öttinger, 1996). Thus, to ensure accurate results, the velocity and stress are obtained by averaging the results of a large number of iterations that were continued after convergence. Furthermore, by using the variance reduction method, the statistical errors have been shown to be acceptable for a mesoscopic approach (Hua and Schieber, 1998, Laso *et al.*, 1997, 1999). In this chapter, the investigation of the statistical error is not carried out for this is typically done in Chapter 4.

7.5.1 10:1 planar contraction flow

This problem was simulated with different models and schemes by several researchers (Feigl and Ottinger, 1996; Laso, 1998; Laso *et al.*, 1999). For the sake of comparison, the same model and physical parameters as in Laso (1998), Laso *et al.* (1999) are used here. In particular, the results for FENE model are compared with the results of Laso (1998), Laso *et al.* (1999).

Owing to the symmetry, only half of the analysis domain needs be considered as shown in Fig. 7.5. Let $2h$ ($h = 0.003mm$) be the height of downstream channel. The height of upstream channel is $2H = 20h$; the upstream and downstream lengths are chosen to be $L_u = 6H$ and $L_d = 60h$, respectively. The fluid parameters are as follows

$$\rho = 1000kg/m^3; \quad \eta_N = 0.01Pa.s; \quad b = 50; \alpha = \frac{\eta_N}{\eta_o} = 0.2; \quad \lambda_H = 2; \quad (7.13)$$

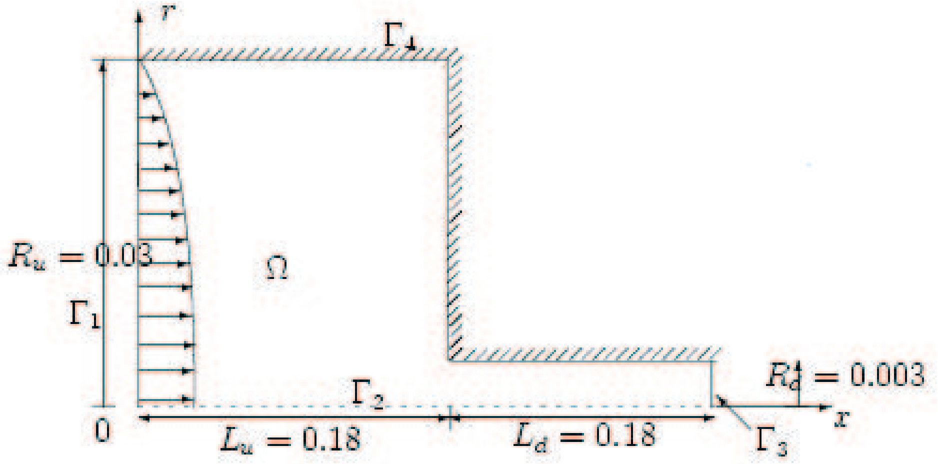


Figure 7.5: Planar 10:1 contraction flow problem: non-slip boundary conditions are imposed at the fluid-solid interfaces; fully developed Poiseuille at the inlet and outlet, and symmetry on the centre-line.

The characteristic length is chosen to be h; the characteristic velocity $\langle u_d \rangle$ (downstream average axial velocity corresponding to the flow rate D_r); the characteristic viscosity $\eta_o = \eta_N + \eta_p$; the characteristic time λ_H and the flow rate is $D_r = 2 \times 10^{-5} m^3/s.m.$

The global boundary conditions of the problem are the same as in Laso *et al.* (1999), Ryssel and Brunn (1999), Raghay and Hakim (2001) where no slip condition is imposed along the wall; a fully developed velocity profile is specified at the inlet and the symmetry condition is imposed on the centre-line.

Let Γ_1, Γ_2, Γ_3 and Γ_4 be the boundaries of the domain Ω (Fig. 7.5). Similar to the boundary conditions described in §5.7.2, here the velocity profile at the inlet is determined from the 2-D Poiseuille flow problem. The inlet velocity profile obtained from this problem (Fig. 7.6) is in good agreement of the results of Laso *et al.* (1999).

The inlet velocity profile for the FENE fluid is slightly different from the

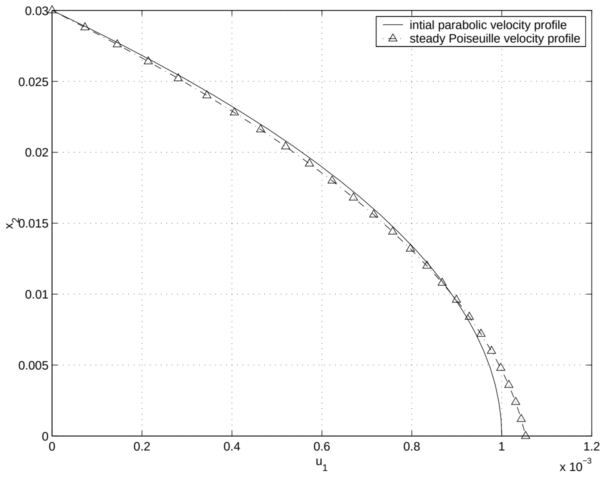


Figure 7.6: The steady state planar 10:1 contraction flow problem using the FENE dumbbell model fluid: the inlet velocity profile obtained from the steady Poiseuille flow problem, $De = 4.4$.

Table 7.1: Planar 10:1 contraction flow problem: four sets of collocation points. N_t: the total number of collocation points; N_{iu}: the number of internal points of the upstream flow; N_{id}: the number of internal points of the downstream flow; N_b: the number of boundary points.

N_t	N_{iu}	N_{id}	N_b
720	453	116	151
1024	667	186	171
1632	1102	312	218
1943	1286	414	243

parabolic profile, which is usually used to approximate the Oldroyd-B fluid. Furthermore, at low Deborah numbers, the exact velocity profile is close to parabolic. Hence, in the case of a small flow rate, a parabolic profile can be used to approximate the inlet velocity profile of a fully developed Poiseuille flows using the FENE model (Bonvin, 2000).

Four different densities of collocation points are employed to simulate this problems and several corresponding geometry parameters are given in Table 7.1. The collocation points are not uniformly distributed. A higher density of collocation points is specified in the reentrant corner area to capture the expected strong flow gradients in that region. The stochastic simulation is time-dependent with simulation time step $\Delta t = 0.005\lambda_H = 0.01$.

In order to deal with this large scale problem, the analysis domain is divided into a number of sub-domains. Each subdomain is governed by a separate CPU. At an iterative step, there are two groups of tasks, one for the stochastic process to determine the stress tensor and the another one is for deterministic operations to calculate the velocity field.

The stochastic task consists of solving SDEs and computing average stresses at collocation points. In general, these tasks are carried out independently, except several data exchange through the interfaces of adjacent subdomains, specially when generating random numbers. In this work, the average stress at the collocation points fixed on interfaces is the average of the whole configuration fields obtained from adjacent sub-domains.

The macroscopic task is to solve the governing PDEs. Owing to the penalty function method, only unknowns $\mathbf{u}$ need to be determined in each sub-domain. The algorithm of the parallel mesh-free TPS-RBFN was presented in §7.3.1 and §7.3.2 in which the sum square error associated with each sub-domain depends on its boundary conditions (i.e. the location of subdomain). For each iterative step, the parallel iterative non-overlapping DD procedure requires the communication between adjacent subdomains such as the interchanges of Dirichlet and Neumann data at their interfaces from the previous iterative step. The results presented in this work correspond to the use of 1632 collocation points with 8 sub-domains and 400 dumbbells assigned at each collocation point. A schematic distribution of collocation points and a sub-regioning with 8 sub-domains are shown in Fig. 7.7.

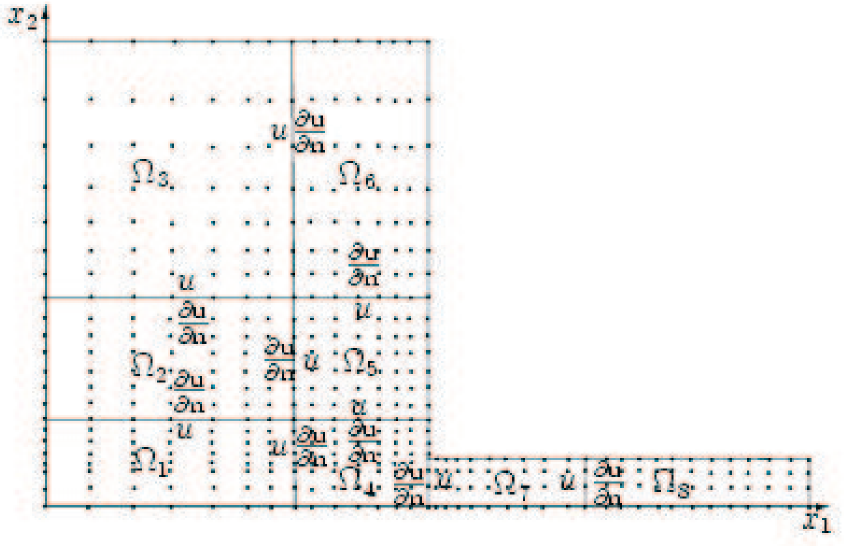

x_2
u
$\frac{\partial u}{\partial n}$
Ω_3
Ω_6
$\frac{\partial u}{\partial n}$
u
u
$\frac{\partial u}{\partial n}$
$\frac{\partial u}{\partial n}$
u
Ω_5
Ω_2
$\frac{\partial u}{\partial n}$
u
$\frac{\partial u}{\partial n}$
u
u
$\frac{\partial u}{\partial n}$
Ω_1
Ω_4
$\frac{\partial u}{\partial n}$
u
Ω_7
u
$\frac{\partial u}{\partial n}$
Ω_8
x_1

Figure 7.7: Planar 10:1 contraction flow problem: Domain decomposition technique using 8 sub-domains; $G_b = [\Omega_1, \Omega_3, \Omega_5, \Omega_7]$ and $G_w = [\Omega_2, \Omega_4, \Omega_6, \Omega_8]$.

Results and discussion

In order to compare the present results with those obtained by Laso *et al.* (1999), the Deborah and Reynold numbers associated with the flow are chosen to be the same, namely,

$$\begin{aligned} De &= \frac{\lambda_H \langle u_d \rangle}{h} = \lambda_H \frac{D_r}{2h^2} = 4.44, \\ Re &= \frac{\rho \langle u_d \rangle h}{\eta_o} = 0.4, \end{aligned}$$

where $\langle u_d \rangle$ is the downstream average axial velocity corresponding to the flow rate D_r. The velocity and stresses are obtained by averaging the results of 500 iterations that were continued after convergence. For this problem the $CM_{\mathbf{u}} < 1.0e{-4}$ for the velocity field (Fig. 7.8) and $CM_{\boldsymbol{\mathcal{T}}} < 2.0e{-3}$ for the stress tensor. Fig. 7.9 shows the streamlines which are in very good agreement with the results of Laso *et al.* (1999) generally. However, a small difference in the vortex shape can be seen. Fig. 7.10 shows the velocity profiles at several cross-sections along the downstream channel $x_1 = 0.18123$ (near the abrupt contraction) and 0.360

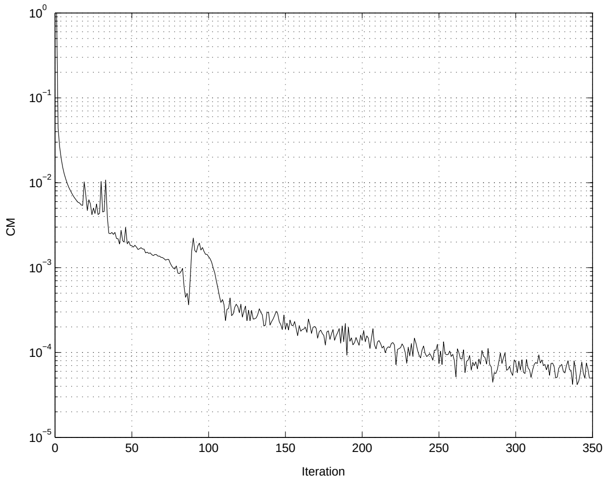


Figure 7.8: The steady state planar 10:1 contraction flow problem using the FENE dumbbell model fluid: the convergence behavior ($CM_{\mathbf{u}}$) of the velocity field $\mathbf{u}$ with respect to the iteration number, using 8 subdomains

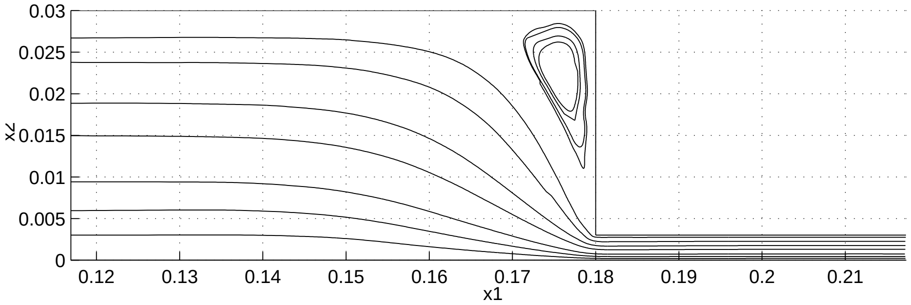


Figure 7.9: The steady state planar 10:1 contraction flow problem using the FENE dumbbell model fluid: the streamline of velocity field, $De = 4.4$.

(outlet). Figure 7.11 shows the variation of the axial velocity on several planes $x_2 = 0.0150$, 0.0028, 0.0013 and 0 (centre-line). The present results generally agree with those obtained by Laso *et al.* (1999), including the velocity overshoot feature at the abrupt contraction, which was also reported in Feigl and Ottinger (1996). Figures 7.12 - 7.13 present the shear stresses along the channel on several planes $x_2 = 0.00020$, 0.00047, 0.00114 and 0.00284. It can be seen that there is an abrupt change of shear stress at the cross-section of the contraction, which is increasingly pronounced towards the reentrant corner. Figs 7.14-7.15 show the first normal stress difference on several planes $x_2 = 0.00020$, 0.00047, 0.00081,

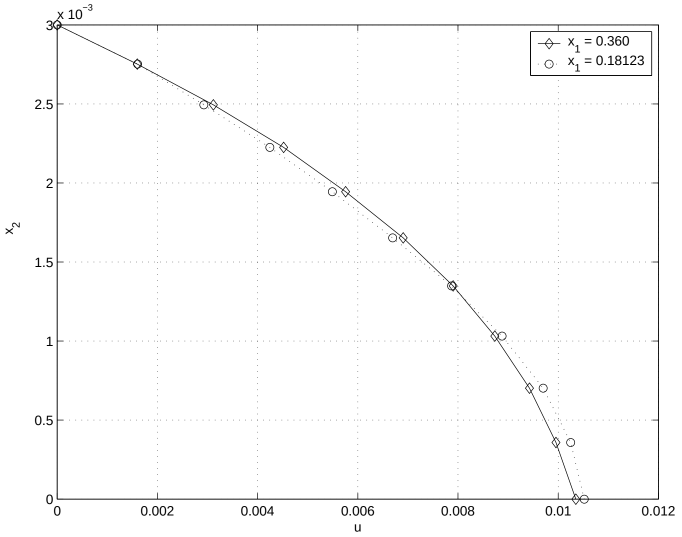


Figure 7.10: The steady state planar 10:1 contraction flow problem using the FENE model fluid: The velocity profile on several cross-sections of the downstream channel $x_1 = 0.360$ (outlet), 0.18123; $De = 4.4$.

0.00114, 0.00258 and 0.00284. On the planes nearer to the centreline, stress overshoots can be observed. Figs 7.16 - 7.17 show the variation in behavior of the polymer shear stresses and the first normal stress differences with Deborah numbers where the strength of the stress overshoots increases with increasing elasticity (higher Deborah number). The problem is also solved using 200 and 500 dumbbells at each collocation points and the results show that there is no discernible difference between using 400 and 500 dumbbells at each collocation points (Fig. 7.18) but an increased oscillation of the stresses when reducing the number of dumbbells at each collocation point down to 200 (Fig. 7.19). Figure 7.20 consolidates the results of shear stresses shown separately in Figs. (7.18) and (7.19) to highlight the oscillatory behavior associated with low numbers of dumbbells.

In order to estimate the efficiency of the parallel algorithm in conjunction with BCF-RBFN mesh-free method, a range of 4, 12, 16 and 20 sub-domains (CPUs) are also carried out using 1632 collocation points. Here, two elements of interest studied to assess this algorithm: (i) the convergence measure or norm of error

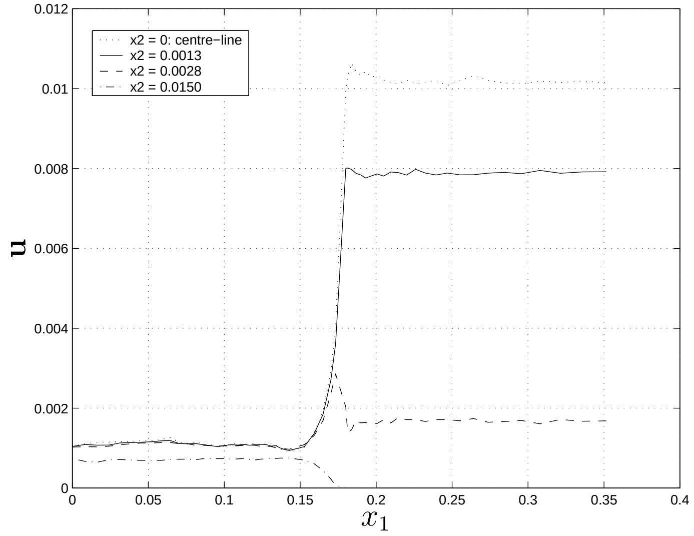


Figure 7.11: The steady state planar 10:1 contraction flow problem using the FENE model fluid: The velocity profile on several planes $x_2 = 0.0013$, 0.0028, 0.0150 and on the centreline; $De = 4.4$.

(CM) and (ii) the efficiency which is defined as the ratio of the execution time using one processor (T1) and the execution time using N processors. For this numerical example, due to the difficulties associated with solving a large scale problem, T1 is obtained from using 8 sub-domains but carrying out serially on only one processor. T1 and the required memory for the problem are 764 minutes and 430 Mbytes, respectively, on Pentium 3, 1.8GHz, using Fortran 90. Table 7.2 shows the effect of the number of sub-domains on the CM and the number of CPUs on the efficiency of the parallel technique. Here, the method of calculating T1 is slightly different from that reported in Tran-Canh and Tran-Cong (2003c).

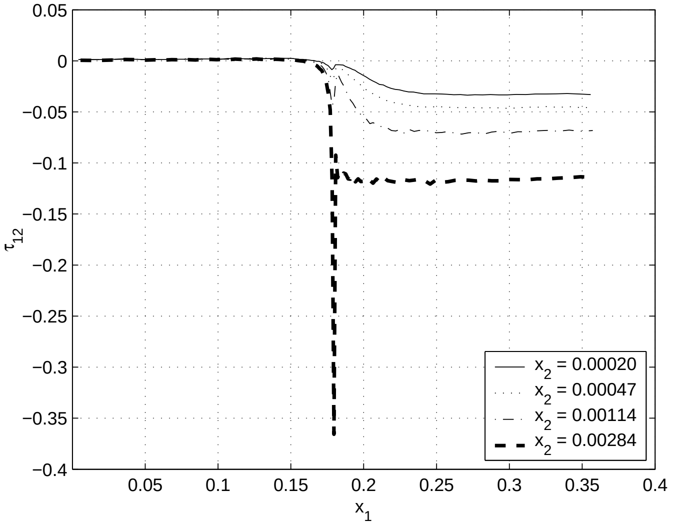


Figure 7.12: The steady state planar 10:1 contraction flow problem using the FENE model fluid: The polymer shear stress on several planes $x_2 = 0.00020$, 0.00047, 0.00114 and 0.00284; $De = 4.4$.

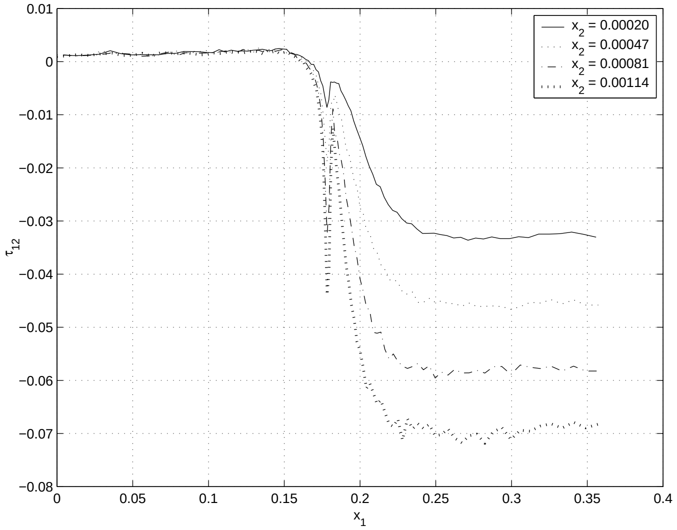


Figure 7.13: The steady state planar 10:1 contraction flow problem using the FENE model fluid: The polymer shear stress on several planes $x_2 = 0.00020$, 0.00047, 0.00081 and 0.00114; $De = 4.4$.

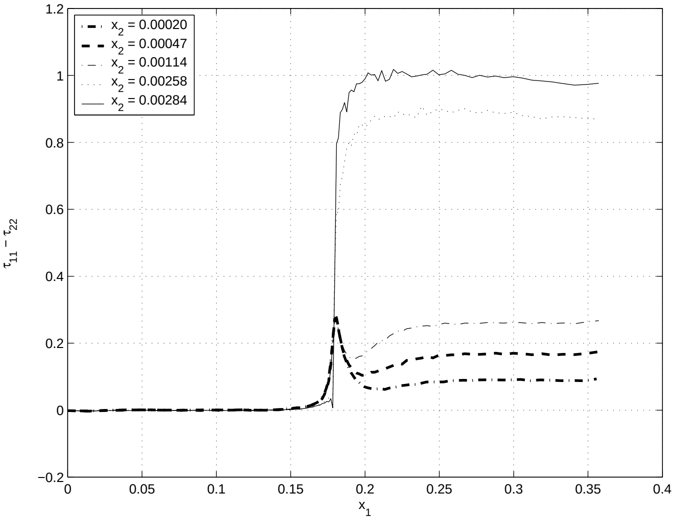


Figure 7.14: The steady state planar 10:1 contraction flow problem using the FENE model fluid: The first polymer normal stress difference on several planes $x_2 = 0.00020$, 0.00047, 0.00114, 0.00258 and 0.00284; $De = 4.4$.

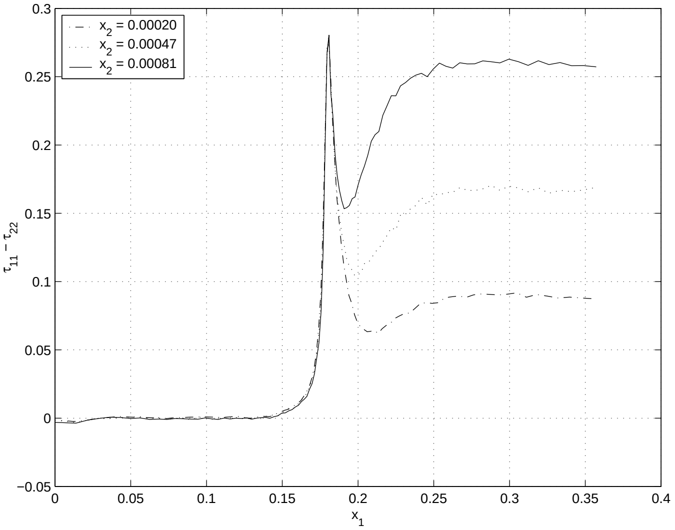


Figure 7.15: The steady state planar 10:1 contraction flow problem using the FENE model fluid: The first polymer normal stress difference on several planes $x_2 = 0.00020$, 0.00047 and 0.00081; $De = 4.4$.

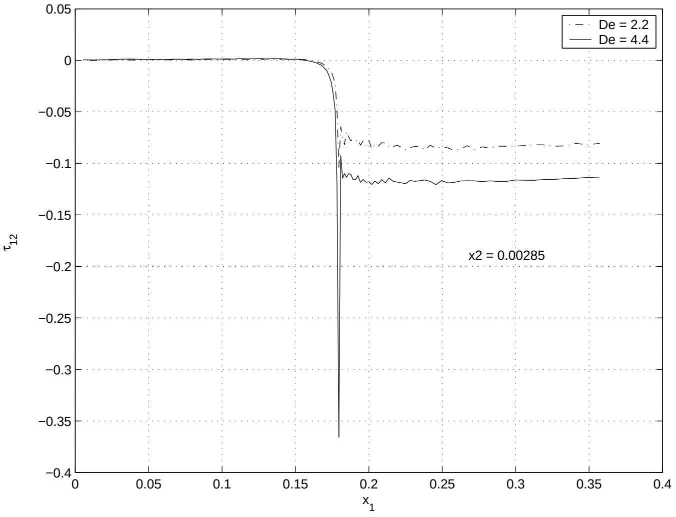


Figure 7.16: The steady state planar 10:1 contraction flow problem using the FENE model fluid: the polymer shear stresses on the plane $x_2 = 0.00285$ near the reentrant corner, for two Deborah numbers $De = 4.4$ and 2.2.

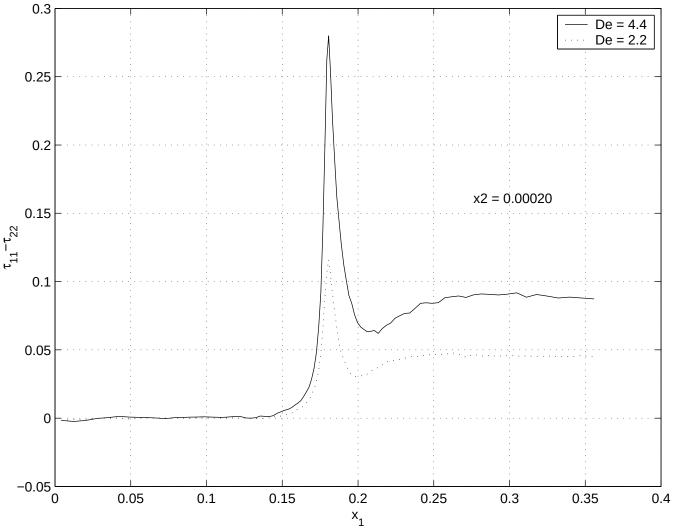


Figure 7.17: The steady state planar 10:1 contraction flow problem using the FENE model fluid: the polymer first normal stress differences on the plane $x_2 = 0.00020$ near the centre-line, for two Deborah numbers $De = 4.4$ and 2.2.

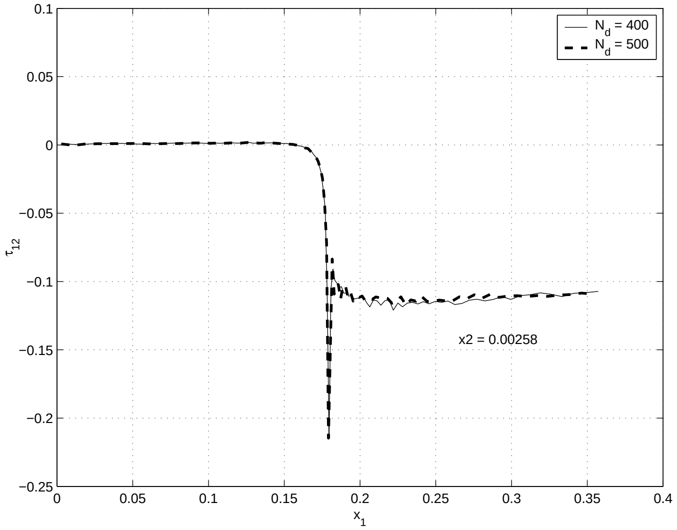


Figure 7.18: The steady state planar 10:1 contraction flow problem using the FENE model fluid: the polymer first normal stress differences on the plane $x_2 = 0.00258$ for Deborah number $De = 4.4$, using $N_d = 400$ and 500 dumbbells at each collocation point.

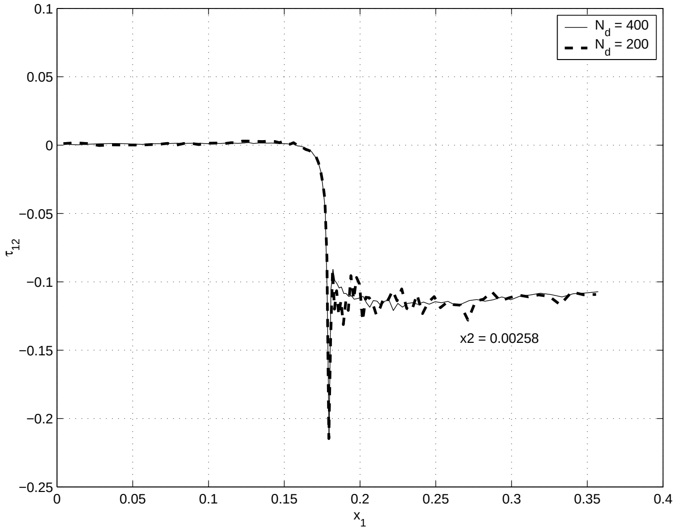


Figure 7.19: The steady state planar 10:1 contraction flow problem using the FENE model fluid: the polymer shear stresses on the plane $x_2 = 0.00258$ for Deborah number $De = 4.4$, using $N_d = 400$ and 200 dumbbells at each collocations.

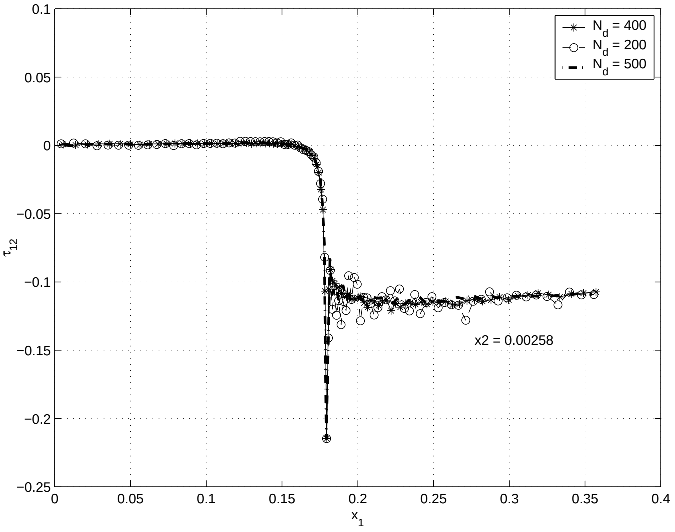


Figure 7.20: The steady state planar 10:1 contraction flow problem using the FENE model fluid: the polymer shear stresses on the plane $x_2 = 0.00258$ for Deborah number $De = 4.4$, using $N_d = 400$, 200 and 500 dumbbells at each collocation point.

Table 7.2: Axisymmetric 10:1 contraction flow problem: two sets of collocation points. N: the number of subdomains; N_t the total number of collocation points; N_s: the number of collocations points of a subdomain; N_{int}: the number of collocation points (range) on the interface of each subdomain; $CM_{\mathbf{u}}$: the convergence measure and E_n: the efficiency coefficient of the algorithm of the parallel DD technique.

N	N_t	N_s	N_{int}	$CM_{\mathbf{u}}$	$E_n\%$
8	1632	210	$10 \rightarrow 51$	$1.12e-4$	87.6
12	1632	140	$10 \rightarrow 48$	$1.20e-4$	83.4
16	1632	110	$10 \rightarrow 42$	$1.82e-4$	77.2
20	1632	90	$10 \rightarrow 38$	$2.12e-4$	68.7

Table 7.3: Lid driven cavity problem: the performance of the parallel domain decomposition with respect to the number of CPUs. N: the number of computing nodes; N_i: the number of collocation points of each sub-domain; N_{int}: the number of the interface points of each sub-domain; $CM_{\mathbf{u}}$: the norm of error and E_n: efficiency of the parallel decomposition scheme.

N	N_i	N_{int}	$CM_{\mathbf{u}}$	E_n
4	441	41	$1.00e-4$	91.4
8	231	31	$1.10e-4$	88.6
16	121	$21 \rightarrow 40$	$1.50e-4$	75.2
20	99	$19 \rightarrow 36$	$1.80e-4$	69.7

The results given in Table 7.2 show that the efficiency coefficient is pretty high in comparison with the results reported on the efficiency of using the parallel techniques in the mesoscopic approach (for example in Laso *et al.*, 1997).

7.5.2 Lid driven cavity flow

In this problem, described in chapter 6, the parallel domain decomposition technique, focuses on both macro and micro components using 41 × 41 collocation points. Some geometrical parameters are given in Table 7.3.

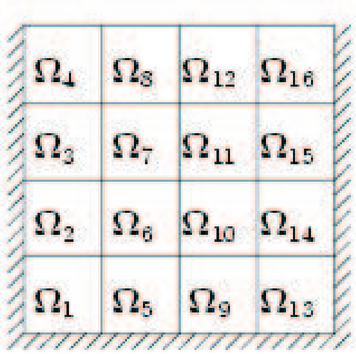

Ω_4 Ω_8 Ω_{12} Ω_{16}
Ω_3 Ω_7 Ω_{11} Ω_{15}
Ω_2 Ω_6 Ω_{10} Ω_{14}
Ω_1 Ω_5 Ω_9 Ω_{13}

Figure 7.21: Lid driven cavity problem: the domain decomposition using 16 sub-domains; $G_b = [\Omega_1, \Omega_3,\ \Omega_5, \Omega_7,\ \Omega_9, \Omega_{11},\ \Omega_{13}, \Omega_{15}]$; $G_w = [\Omega_2, \Omega_4,\ \Omega_6, \Omega_8,\ \Omega_{10}, \Omega_{12},\ \Omega_{14}, \Omega_{16}]$; Dirichlet boundary conditions are imposed on interfaces for G_b and Neumann boundary conditions are for G_w.

The efficiency of the algorithm is estimated using 4, 8, 16, 20 subdomains and the task of each subdomain is carried out independently by a separate processor. For this example, the collocation points are uniformly distributed and subdomains have the same size (Fig. 7.21).

The fluid parameters are given in §6.5.3. Since many results of the problem was described in chapter 6, here, only results concerning the performance of the parallel DD techniques are presented. Although the convergence measure of the scheme is affected by the number of subdomains (and CPUs) (Table 7.3), the results of velocity fields using 4, 8, 16, 20 sub-domains, given in Fig. 7.22, show that the differences are not significant.

Fig. 7.23 depicts the convergence behavior of the problem using 8 sub-domains. The performance of the algorithm of the parallel technique given in table 7.3 shows that the efficiency of the parallel technique is pretty high for a mesoscopic method. Here, the computation time T1 and required memory, using (41x41+2) collocation points are 733 minutes and 441 Mbytes, respectively, on Pentium 3, 1.8GHz, using Fortran 90.

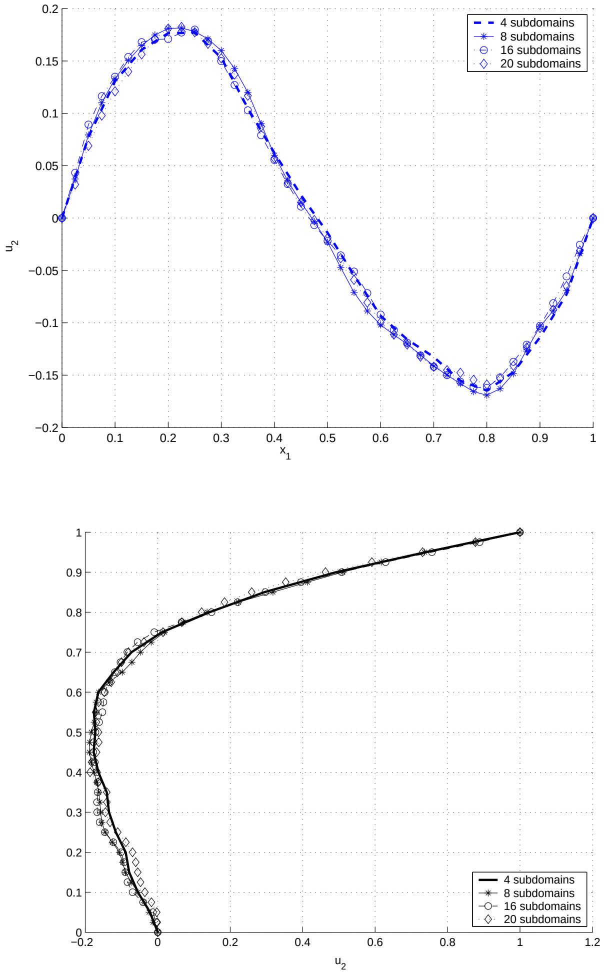


Figure 7.22: The lid driven square cavity flow problem using the Hookean dumbbell model fluid: the profile of the velocity component u_2 on the horizontal central plane (lower figure) and the profile of the velocity component u_1 on the vertical central plane (upper figure) using parallel domain decomposition techniques. '$--$' for 4 sub-domains, '$- * -$' for 8 sub-domains, '$-o-$' for 16 sub-domains and $..\diamond$ for 20 sub-domains. The parameters are $\alpha = 1/9$, $\lambda_H = 1 \quad (De = 1)$, $(41 \times 41 + 2)$ collocation points; $De = 1$.

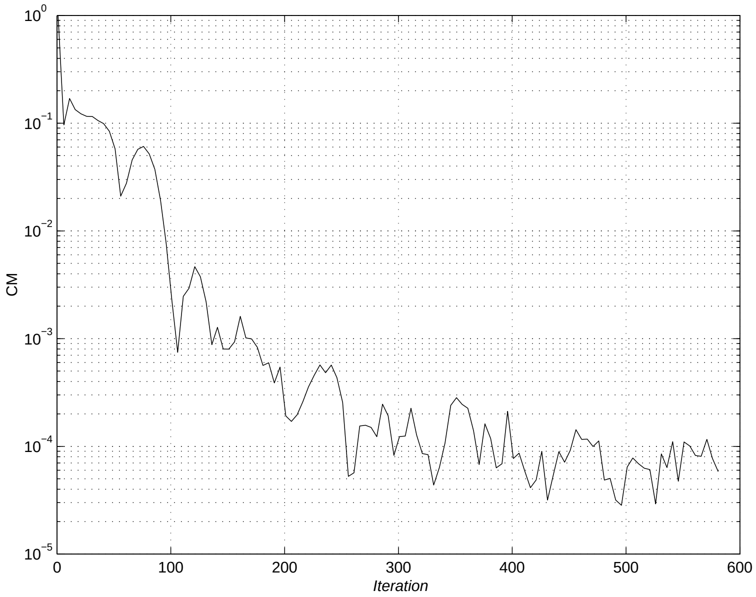


Figure 7.23: The lid driven cavity using the Hookean dumbbell model fluid: the convergence behavior ($CM_{\mathbf{u}}$) of the velocity field $\mathbf{u}$ with respect to the iteration number, using 8 subdomains

7.6 Concluding remarks

A parallel domain decomposition mesh-free TPS-RBFN-BCF method for dilute polymer solution is reported. The incorporation of a parallel domain decomposition technique makes the RBFN-BCF element-free methods more suitable for large scale problems. The iterative non-overlapping domain decomposition method, which is employed in both macro and micro components of the hybrid TPS-RBFN-BCF simulation scheme, converts a large problem into a number of smaller ones, facilitates coarse grained parallelization and decreases the wall time of a micro-macroscopic method. The experience gained from solving some benchmark problems shows the following (i) Although the convergence of the scheme can be affected by the number of subdomains, the results obtained are very good, judging by the convergence measure (Tables 7.2 and 7.3); (ii) In spite of the complexity involved in parallelizing the mesoscopic approach as mentioned in Laso *et al.* (1997), the achieved efficiency is high (Tables 7.2 and

7.3) ($\geq 60\%$ when using 20 CPUs); (iii) It is relatively easy and simple to implement parallel domain decomposition using MPI for both macro and microscopic components.

Chapter 8

Conclusion

With the aim to create a new and improved numerical approach for the analysis of non-Newtonian fluid flows, the objectives of the present research effort have been attained through a number of logical steps, progressing from meshless function approximation, partial removal of element-based discretisation to total element-free simulation methods.

The proven "universal approximation" capability of neural-like networks, such as the MLPNs and the RBFNs, is the theoretical basis for the element-free approach to function approximation and the numerical solution of PDEs. This numerical approach is relatively new, particularly in relation to the numerical solution of PDEs. In the context of non-Newtonian fluid flow, the present endeavour is a significant contribution of new knowledge in the application of the meshless methodology to the hybrid simulation approach which is coming from the CONNFFESSIT philosophy. The least-squares point collocation technique eliminates element-based discretisation and the SST renders closed form constitutive equations unnecessary. The system matrix size is reduced by the decoupling of the computation of the velocity field (macroscopic component) from the calculation of the stress field (microscopic component). At a more detailed level, various techniques are introduced to make the overall approach more effective and efficient. These improvements include the use of penalty function in the macroscopic component, the velocity gradient-based domain decomposition, the variance reduction, network regularisation and the iterative non-overlapping domain decomposition techniques.

Apart from the improvement of the hybrid simulation approach based on Lagrangian SST, the present element-free methodology is particularly helpful in the parallelisation of the Eulerian hybrid simulation method. Domain decomposition (iterative non-overlapping), which is relatively simple in the element-free framework, helps remove the matrix ill-condition problems in larger problems. Thus, in its final form, the present method is not only simpler to implement, but also more efficient and effective. The element-free nature of the method makes its application much easier in situations where complex geometries or moving boundaries are involved.

The methods have been successfully implemented with appropriate verification using

- circular Couette flow of GNFs.
- start-up planar Couette flow of FENE and FENE-P fluids;
- planar Poiseuille flow of Hookean, FENE and FENE-P fluids.

The effectiveness and efficiency of the methods have been established with the simulation of non-trivial problems, including

- axisymmetric 4:1 contraction flow of Hookean fluids. In this case the circular Poiseuille flow was simulated and the results were used in specifying the inlet condition;
- planar 10:1 contraction flow of FENE and FENE-P fluids. In this case the planar Poiseuille flow was simulated and the results were used in specifying the inlet condition;

- lid-driven cavity flow of Hookean fluids.

For the above problems, some outcome highlights include

- convergence has been achieved for the first time at very low value of the power law index;
- comparable or better accuracy has been achieved with coarser discretisation, both spatially and temporally;
- highly parallelised implementation.

Finally, some outstanding issues for further investigation are

- Free surface flow simulation;
- Flow analysis with more realistic models of complex fluids (for example, the inclusion of hydrodynamic interaction in the dumbbell models for dilute polymer solutions, and Reptation models for concentrated solution and melts);
- Approximation of SDE's (microscopic component) using neural networks-based methods;
- Overlapping domain decomposition techniques.

Appendix A

Complements to integral equations

A.1 Fundamental solution (Kevin solution)

The small displacement field (Kevin solution) which is produced by a concentrated force $\mathbf{f}$ at a point $(\mathbf{y})$ in a homogenous, isotropic and infinite linear elastic medium is given by

$$u_i(\mathbf{x}) = u_{ij}^*(\mathbf{x}, \mathbf{y}) f_j(\mathbf{y}), \tag{A.1}$$

where

$$\begin{aligned} u_{ij}^*(\mathbf{x}, \mathbf{y}) &= \frac{1}{8\pi(1-\nu)\eta}\left[(3-4\nu)\delta_{ij}\ln(r) + \frac{r_i r_j}{r^2}\right] \quad \text{for} \quad \text{2D-space}, \\ u_{ij}^*(\mathbf{x}, \mathbf{y}) &= \frac{1}{16\pi(1-\nu)\eta r}\left[(3-4\nu)\delta_{ij} + \frac{r_i r_j}{r^2}\right] \quad \text{for} \quad \text{3D-space}, \end{aligned} \tag{A.2}$$

$r = \|\mathbf{r}\| = \|\mathbf{x} - \mathbf{y}\|$, δ_{ij} is the Kronecker delta, η is the shear modulus and ν is the Poisson's ratio.

The associated traction is given by

$$t_i(\mathbf{x}) = t_{ij}^*(\mathbf{x}, \mathbf{y}) f_j(\mathbf{y}), \tag{A.3}$$

$$t^*_{ij}(\mathbf{x},\mathbf{y}) = -\frac{1}{4\alpha\pi(1-\nu)r^\alpha}\left\{\left[(1-2\nu)\delta_{ij} + \beta\frac{r_i r_j}{r^2}\right]\frac{\partial r}{\partial n} - (1-2\nu)\left(\frac{r_i}{r}n_j - \frac{r_j}{r}n_i\right)\right\}, \quad \text{(A.4)}$$

$\alpha = 1, 2$ and $\beta = 2, 3$ are for two and three-dimensional problems, respectively and $\mathbf{n}(\mathbf{x})$ is the outward unit normal vector at a point $\mathbf{x}$.

If the material is an infinite medium of incompressible viscous fluid, the corresponding fundamental solution (Stokeslet) and its associated traction field are obtained from (A.2) and (A.4) by interpreting $\mathbf{u}$ as velocity, η as viscosity and $\nu = 0.5$.

A.2 Analytical integration of functions involving 2D kernels

In this appendix, superscripts are used to denote the node of the element while subscripts denote tensor components (lower case symbols, e.g. a_{ij}) or simply identifiers for upper case symbols (e.g. C_1). Consider the following integrals in standard BIE formulation

$$I = \int_{\Gamma_e} u^*_{ij}(\mathbf{x},\mathbf{y})t_j(\mathbf{y})d\Gamma(\mathbf{y}) \quad \text{(A.5)}$$

$$J = \int_{\Gamma_e} t^*_{ij}(\mathbf{x},\mathbf{y})u_j(\mathbf{y})d\Gamma(\mathbf{y}) \quad \text{(A.6)}$$

where Γ_e is the boundary element defined by a straight line segment between two nodes 1 and 2 as shown in Figure A.1; $\mathbf{y} \in \Gamma_e$ is the field point; $\mathbf{x}$ is the source point under consideration; $u^*_{ij}(\mathbf{x},\mathbf{y})$ is the i component of velocity field at $\mathbf{x}$ due to a point force in j direction at $\mathbf{y}$ (Kelvin fundamental solution) and $t^*_{ij}(\mathbf{x},\mathbf{y})$ is its associated traction; $u_j(\mathbf{y})$ and $t_j(\mathbf{y})$ are the j components of velocity and boundary traction at $\mathbf{y}$, respectively. In order to estimate (A.5) and (A.6), the components $u_j(\mathbf{y})$ and $t_j(\mathbf{y})$ can be assumed to have a certain shape (usually constant, linear or quadratic). In this work, the linear shape function is employed and hence the components $u_j(\mathbf{y})$ and $t_j(\mathbf{y})$ are expressed as

$$\begin{aligned} u_j(\mathbf{y}) &= N_1(\xi)u_j^{(1)} + N_2(\xi)u_j^{(2)} && \text{(A.7)} \\ t_j(\mathbf{y}) &= N_1(\xi)t_j^{(1)} + N_2(\xi)t_j^{(2)} && \text{(A.8)} \\ y_j &= N_1(\xi)y_j^{(1)} + N_2(\xi)y_j^{(2)} && \text{(A.9)} \end{aligned}$$

where $u_j^{(1)}$, $t_j^{(1)}$, $y_j^{(1)}$ and $u_j^{(2)}$, $t_j^{(2)}$, $y_j^{(2)}$ are the j components of the nodal velocity, nodal traction and coordinate associated with nodes 1 and 2, respectively; N_1 and N_2 are known linear shape functions and ξ is the natural coordinate ($-1 \leq \xi \leq 1$). Substitution of (A.8) into (A.5) and (A.7) into (A.6) yields

$$\begin{aligned} I &= \int_{\Gamma_e} u_{ij}^*(\mathbf{x},\mathbf{y}) \left(N_1(\xi(\mathbf{y}))t_j^{(1)} + N_2(\xi(\mathbf{y}))t_j^{(2)} \right) d\Gamma(\mathbf{y}) \\ &= t_j^{(1)} \int_{\Gamma_e} N_1(\xi(\mathbf{y}))u_{ij}^*(\mathbf{x},\mathbf{y})d\Gamma(\mathbf{y}) + t_j^{(2)} \int_{\Gamma_e} N_2(\xi(\mathbf{y}))u_{ij}^*(\mathbf{x},\mathbf{y})d\Gamma(\mathbf{y}) \quad \text{(A.10)} \end{aligned}$$

$$\begin{aligned} J &= \int_{\Gamma_e} t_{ij}^*(\mathbf{x},\mathbf{y}) \left(N_1(\xi(\mathbf{y}))u_j^{(1)} + N_2(\xi(\mathbf{y}))u_j^{(2)} \right) d\Gamma(\mathbf{y}) \\ &= u_j^{(1)} \int_{\Gamma_e} N_1(\xi(\mathbf{y}))t_{ij}^*(\mathbf{x},\mathbf{y})d\Gamma(\mathbf{y}) + u_j^{(2)} \int_{\Gamma_e} N_2(\xi(\mathbf{y}))t_{ij}^*(\mathbf{x},\mathbf{y})d\Gamma(\mathbf{y}) \quad \text{(A.11)} \end{aligned}$$

By denoting

$$\begin{aligned} g_{ij}^{(1)} &= \int_{\Gamma_e} N_1(\xi(\mathbf{y}))u_{ij}^*(\mathbf{x},\mathbf{y})d\Gamma(\mathbf{y}) = \int_{\Gamma_e} N_1(\xi)u_{ij}^*(\mathbf{x}(\xi),\mathbf{y}(\xi))\frac{l}{2}d\xi \text{(A.12)} \\ g_{ij}^{(2)} &= \int_{\Gamma_e} N_2(\xi(\mathbf{y}))u_{ij}^*(\mathbf{x},\mathbf{y})d\Gamma(\mathbf{y}) = \int_{\Gamma_e} N_2(\xi)u_{ij}^*(\mathbf{x}(\xi),\mathbf{y}(\xi))\frac{l}{2}d\xi \text{(A.13)} \\ h_{ij}^{(1)} &= \int_{\Gamma_e} N_1(\xi(\mathbf{y}))t_{ij}^*(\mathbf{x},\mathbf{y})d\Gamma(\mathbf{y}) = \int_{\Gamma_e} N_1(\xi))t_{ij}^*(\mathbf{x}(\xi),\mathbf{y}(\xi))\frac{l}{2}d\xi \text{(A.14)} \\ h_{ij}^{(2)} &= \int_{\Gamma_e} N_2(\xi(\mathbf{y}))t_{ij}^*(\mathbf{x},\mathbf{y})d\Gamma(\mathbf{y}) = \int_{\Gamma_e} N_2(\xi)t_{ij}^*(\mathbf{x}(\xi),\mathbf{y}(\xi))\frac{l}{2}d\xi, \text{(A.15)} \end{aligned}$$

(A.10) and (A.11) become

$$\begin{aligned} I &= t_j^{(1)}g_{ij}^{(1)} + t_j^{(2)}g_{ij}^{(2)} && \text{(A.16)} \\ J &= u_j^{(1)}h_{ij}^{(1)} + u_j^{(2)}h_{ij}^{(2)} && \text{(A.17)} \end{aligned}$$

The integrals $g_{ij}^{(1)}$, $g_{ij}^{(2)}$, $h_{ij}^{(1)}$ and $h_{ij}^{(2)}$ can be obtained analytically when the source point $\mathbf{x}$ lies on the line through nodes 1 and 2 and the final results are given as follows

- $\mathbf{x} \equiv \mathbf{x}_1$

$$\begin{aligned} g_{ij}^{(1)} &= \frac{C_1 l}{2}\left[a_{ij} + C_2\delta_{ij}\left(\frac{3}{2} - \ln(l)\right)\right] && \text{(A.18)} \\ g_{ij}^{(2)} &= \frac{C_1 l}{2}\left[a_{ij} + C_2\delta_{ij}\left(\frac{1}{2} - \ln(l)\right)\right] && \text{(A.19)} \\ h_{ij}^{(2)} &= b_{ij}C_3C_4 && \text{(A.20)} \end{aligned}$$

Note that $h_{ij}^{(1)}$ can be obtained by using the condition of rigid body motion.

- $\mathbf{x} \equiv \mathbf{x}_2$

$$\begin{aligned} g_{ij}^{(1)} &= \frac{C_1 l}{2}\left[a_{ij} + C_2\delta_{ij}\left(\frac{1}{2} - \ln(l)\right)\right] && \text{(A.21)} \\ g_{ij}^{(2)} &= \frac{C_1 l}{2}\left[a_{ij} + C_2\delta_{ij}\left(\frac{3}{2} - \ln(l)\right)\right] && \text{(A.22)} \\ h_{ij}^{(1)} &= -b_{ij}C_3C_4 && \text{(A.23)} \end{aligned}$$

Note that $h_{ij}^{(2)}$ can be obtained by using the condition of rigid body motion.

- $\mathbf{x} \equiv \mathbf{p}$

$$\begin{aligned} g_{ij}^{(1)} = {}& \frac{C_1 l a_{ij}}{2} - \frac{C_1C_2D_2\delta_{ij}}{l}\left(D_2\ln(D_2) - D_1\ln(D_1) - l\right) \\ &+ \frac{C_1C_2\delta_{ij}}{2l}\left(D_2^2\ln(D_2) - D_1^2\ln(D_1) - Dl\right) && \text{(A.24)} \end{aligned}$$

$$\begin{aligned} g_{ij}^{(2)} = {}& \frac{C_1 l a_{ij}}{2} + \frac{C_1C_2D_1\delta_{ij}}{l}\left(D_2\ln(D_2) - D_1\ln(D_1) - l\right) \\ &- \frac{C_1C_2\delta_{ij}}{2l}\left(D_2^2\ln(D_2) - D_1^2\ln(D_1) - Dl\right) && \text{(A.25)} \end{aligned}$$

$$\begin{aligned} h_{ij}^{(1)} &= \frac{C_3C_4b_{ij}}{l}\left(D_2(\ln(D_2) - \ln(D_1)) - l\right) && \text{(A.26)} \\ h_{ij}^{(2)} &= -\frac{C_3C_4b_{ij}}{l}\left(D_1(\ln(D_2) - \ln(D_1)) - l\right) && \text{(A.27)} \end{aligned}$$

- $\mathbf{x} \equiv \mathbf{q}$

$$\begin{aligned} g_{ij}^{(1)} = {}& \frac{C_1 l a_{ij}}{2} + \frac{C_1C_2D_1\delta_{ij}}{l}\left(D_2\ln(D_2) - D_1\ln(D_1) - l\right) \\ &- \frac{C_1C_2\delta_{ij}}{2l}\left(D_2^2\ln(D_2) - D_1^2\ln(D_1) - Dl\right) && \text{(A.28)} \end{aligned}$$

$$g_{ij}^{(2)} = \frac{C_1 l a_{ij}}{2} - \frac{C_1 C_2 D_2 \delta_{ij}}{l}\left(D_2 \ln(D_2) - D_1 \ln(D_1) - l\right) + \frac{C_1 C_2 \delta_{ij}}{2l}\left(D_2^2 \ln(D_2) - D_1^2 \ln(D_1) - Dl\right) \tag{A.29}$$

$$h_{ij}^{(1)} = \frac{C_3 C_4 b_{ij}}{l}\left(D_1(\ln(D_2) - \ln(D_1)) - l\right) \tag{A.30}$$

$$h_{ij}^{(2)} = -\frac{C_3 C_4 b_{ij}}{l}\left(D_2(\ln(D_2) - \ln(D_1)) - l\right) \tag{A.31}$$

- $\mathbf{x} \equiv \mathbf{r}$: This case corresponds to the special treatment of the corner situation where the traction is two-valued. Following the technique devised by Tran-Cong and Phan-Thien (1988) for a general resolution of this problem, the traction at the corner is assumed to have different value for each element connecting to the corner node. Then each element is assigned a collocation point (i.e. $\mathbf{r}$) situated on the element but slightly shifted from the corner to provide the sufficient number of algebraic equations for the unknowns.

$$g_{ij}^{(1)} = \frac{C_1 D_1}{2}\left[a_{ij} + C_2 \delta_{ij}\left(\frac{1}{2} - \ln(D_1)\right)\right] + \frac{N_1(\xi) C_1 l}{2}\left[a_{ij} + C_2 \delta_{ij}\left(\frac{3}{2} - \frac{D_1 \ln(D_1)}{l} - \frac{D_2 \ln(D_2)}{l}\right)\right] \tag{A.32}$$

$$g_{ij}^{(2)} = \frac{C_1 D_2}{2}\left[a_{ij} + C_2 \delta_{ij}\left(\frac{1}{2} - \ln(D_2)\right)\right] + \frac{N_2(\xi) C_1 l}{2}\left[a_{ij} + C_2 \delta_{ij}\left(\frac{3}{2} - \frac{D_1 \ln(D_1)}{l} - \frac{D_2 \ln(D_2)}{l}\right)\right] \tag{A.33}$$

$$h_{ij}^{(1)} = \frac{b_{ij} C_3 C_4}{2}\left[(1 - \xi)\left(\ln(D_2) - \ln(D_1)\right) - 2\right] \tag{A.34}$$

$$h_{ij}^{(2)} = \frac{b_{ij} C_3 C_4}{2}\left[(1 + \xi)\left(\ln(D_2) - \ln(D_1)\right) - 2\right] \tag{A.35}$$

in which ξ is determined from $\mathbf{x} = N_1(\xi)\mathbf{y}^{(1)} + N_2(\xi)\mathbf{y}^{(2)}$

where

$$l_i = y_i^{(2)} - y_i^{(1)} \tag{A.36}$$

$$a_{ij} = \frac{l_i l_j}{l^2} \tag{A.37}$$

$$n_1 = \frac{l_2}{l}, \quad n_2 = -\frac{l_1}{l} \tag{A.38}$$

$$b_{ij} = \frac{l_i n_j - l_j n_i}{l} \tag{A.39}$$

$$C_1 = \frac{1}{8G(1-\nu)\pi}, \quad G = \frac{E}{2(1+\nu)} \tag{A.40}$$

$$C_2 = 3 - 4\nu \tag{A.41}$$

$$C_3 = \frac{1}{4\pi(1-\nu)} \tag{A.42}$$

$$C_4 = 1 - 2\nu \tag{A.43}$$

$$D_1 = \sqrt{(\mathbf{x} - \mathbf{x}_1)^T(\mathbf{x} - \mathbf{x}_1)} \tag{A.44}$$

$$D_2 = \sqrt{(\mathbf{x} - \mathbf{x}_2)^T(\mathbf{x} - \mathbf{x}_2)} \tag{A.45}$$

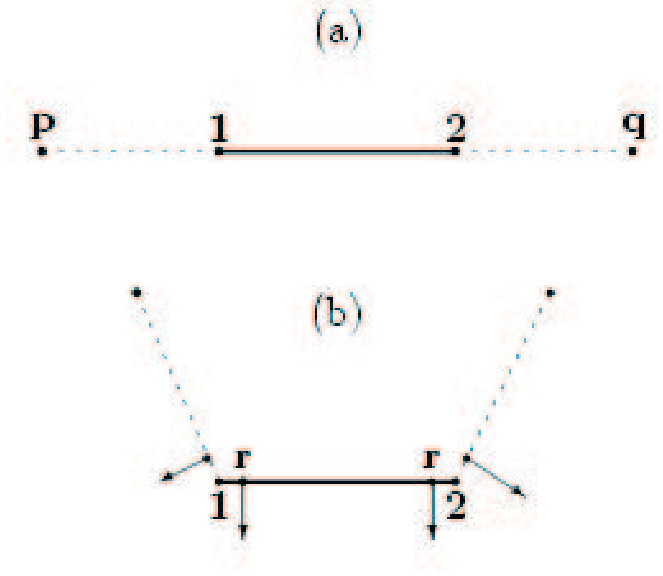


Figure A.1: Integration of functions involving the kernels over the boundary element defined by a straight line segment between two nodes 1 ($\mathbf{x}_1$) and 2 ($\mathbf{x}_2$). Neighbouring elements are denoted by dashed lines. Nodes 1 and 2 are normal nodes in figure (a) and corner nodes in figure (b). In the latter case, due to the existence of two traction vectors associated with the two elements at the same corner, two collocation points per node are located at $\mathbf{r}$ away from the corner.

Appendix B

Complements to FFNNs

B.1 Back Propagation (BP) and Levenberg Marquardt algorithms

The MLPN can be trained to approximate a function over a domain and the MLPN training is considered as an optimization problem in the weight space. The mean square error (MSE) is computed for a MLPN relative to the desire output as follows

$$MSE(\mathbf{w}) = \sum_{i=1}^{m} e_i^2 = \sum_{i=1}^{m} (\hat{y}_i - f(\mathbf{x}_i))^2, \tag{B.1}$$

where $\hat{y}_i$'s are the desired outputs and $\{\mathbf{x}_i, \hat{y}_i\}_{i=1}^{m}$ are data patterns; $f(\mathbf{x}_i)$s are approximated outputs corresponding to the inputs $\mathbf{x}_i$s; $\mathbf{w} = [w_1, w_2, ..., w_n]$ is the weight vector and $\mathbf{e} = [e_1, e_2, ..., e_i, ..., e_m]$ is the error vector.

The BP training consists of the following steps

- An input pattern $\{\mathbf{x}_i, y_i\}_{m}^{i=1}$ sweeps forward through the network to generate an output $f(\mathbf{x}_i)$. Hence, BP starts a loop over all training pattern and computes the corresponding errors by feeding the input forward through the network;

- The mean square error is determined via the equation (B.1);
- An optimization algorithm based on the backward propagation of errors is employed to adjust the values of the weights.

Among the optimization algorithms such as the Adaptive learning rate (Jacob, 1988); Newton and Quasi Newton methods (Becker and Le cun, 1989); Gauss-Newton method (Mckeown *et al.*, 1997), Leveberg-Marquardt algorithm (Levenberg, 1994), the latter one is considered as an efficient method for minimizing the MSE whose weight vector is updated as

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \Delta\mathbf{w} \tag{B.2}$$

$$\Delta\mathbf{w} = -(\mathbf{J}^T\mathbf{J} + \mu\mathbf{I})^{-1}\mathbf{J}^T\mathbf{e} \tag{B.3}$$

where $\mathbf{w}_{t+1}$ is the weight vector at the time $(t+1)$; $\mathbf{w}_t$ is the weight vector at the current time; $\mathbf{e}$ is the current error vector; $J = \frac{\partial\mathbf{e}}{\partial\mathbf{w}}$ is the Jacobian matrix of $\mathbf{e}$; μ is a scalar; $\mathbf{I}$ is the identity matrix. Depending on the value of μ, the Levenberg-Marquardt recovers the steepest decent scheme if μ is very large or Gauss-Newton method in the case of small μ. The line searches have been employed to improve convergence. This is a powerful technique when using L-M (Levenberg-Marquardt) method. In the present method, line searches are used to determine the value of μ in Eq. (B.3) and then the step size is adjusted accordingly in an iteration where the CM does not significantly reduce in comparison with the CM in the previous iteration.

B.2 Least-squares method for the optimisation of weight vectors

In the case of least-squares applied to RBFN models, the function to be minimized with regularization is the sum-square error $C(\mathbf{w}, \boldsymbol{\lambda})$ (3.39) as follows

$$C(\mathbf{w}, \boldsymbol{\lambda}) = \sum_{i=1}^{n}(\hat{y}_i - f(\mathbf{x}_i))^2 + \sum_{j=1}^{m} \lambda_j (w^j)^2, \tag{B.4}$$

where in general $\{\lambda_j\}_{j=1}^{m}$ is the non-negative local regularization parameters and the second term of (B.4) is called the additional weight penalty term; $f(\mathbf{x}) = \sum_{j=1}^{m} w^j h^j(\mathbf{x})$. $\lambda_j = 0, \forall j$ is the case of non-regularization. Differentiating the cost function (B.4) for each j^{th} weight, we have

$$\frac{\partial C}{\partial w^j} = 2\sum_{i=1}^{n}(f(\mathbf{x}_i) - \hat{y}_i)\frac{\partial f}{\partial w^j}(\mathbf{x}_i) + 2\lambda_j w^j; \tag{B.5}$$

where $\frac{\partial f}{\partial w^j}(\mathbf{x}_i) = h^j(\mathbf{x}_i)$. Substituting this derivative into (B.5), the minimization of (B.4) leads to the following equations

$$\sum_{i=1}^{n} f\left(\mathbf{x}_i\right) h^j\left(\mathbf{x}_i\right) + \lambda_j w^j = \sum_{i=1}^{n} \hat{y}_i h^j\left(\mathbf{x}_i\right) \quad j = 1, ..., m. \tag{B.6}$$

Each equation corresponding to j represents a constraint on the solution. In general, the system of equations (B.6) has a unique solution because the number of unknowns and constraints are the same. This system of m equations is rewritten via the language of matrices as follows

$$\mathbf{H}^T\mathbf{f} + \boldsymbol{\lambda}\mathbf{w} = \mathbf{H}^T\hat{\mathbf{y}}; \tag{B.7}$$

where

$$\boldsymbol{\lambda} = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_m \end{bmatrix}; \quad \mathbf{f} = \begin{bmatrix} f(\mathbf{x}_1) \\ f(\mathbf{x}_2) \\ \vdots \\ f(\mathbf{x}_n) \end{bmatrix}; \quad \hat{\mathbf{y}} = \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_n \end{bmatrix};$$

and

$$\mathbf{H} = \begin{bmatrix} h^1(\mathbf{x}_1) & h^2(\mathbf{x}_1) & \cdots & h^m(\mathbf{x}_1) \\ h^1(\mathbf{x}_2) & h^2(\mathbf{x}_2) & \cdots & h^m(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ h^1(\mathbf{x}_n) & h^2(\mathbf{x}_n) & \cdots & h^m(\mathbf{x}_n) \end{bmatrix}. \tag{B.8}$$

$\mathbf{f}$ can be decomposed into the product of two terms, the design matrix and the weight vector as follows

$$\mathbf{f} = \mathbf{H}\mathbf{w}. \tag{B.9}$$

Substituting (B.9) into the equation (B.7), we have

$$\begin{aligned} \mathbf{H}^T\hat{\mathbf{y}} &= \mathbf{H}^T\mathbf{f} + \boldsymbol{\lambda}\mathbf{w} && \text{(B.10)} \\ &= \mathbf{H}^T\mathbf{H}\mathbf{w} + \boldsymbol{\lambda}\mathbf{w} && \text{(B.11)} \\ &= \left(\mathbf{H}^T\mathbf{H} + \boldsymbol{\lambda}\right)\mathbf{w}. && \text{(B.12)} \end{aligned}$$

There are two cases

(1.) For the global regularization ($\lambda_i = \lambda, \forall i$), (B.12) becomes

$$\left(\mathbf{H}^T\mathbf{H} + \lambda\mathbf{I}\right)\mathbf{w} = \mathbf{H}^T\hat{\mathbf{y}}, \tag{B.13}$$

where $\mathbf{I}$ is unity matrix;

(2.) For the case where there is no regularization ($\boldsymbol{\lambda} = \mathbf{0}$), (B.12) is rewritten

$$\left(\mathbf{H}^T\mathbf{H}\right)\mathbf{w} = \mathbf{H}^T\hat{\mathbf{y}} \tag{B.14}$$

B.3 The variance matrix

Since the weights have been determined from the basis of a measurement, $\hat{\mathbf{y}}$, of a stochastic variable $\mathbf{y}$, the corresponding uncertainty in $\mathbf{w}$ needs to be known and depends on the nature of $\hat{\mathbf{y}}$ and the relationship between $\mathbf{w}$ and $\mathbf{y}$. In this work, the noise affecting $\mathbf{y}$ is considered to be normal and independently, identically distributed

$$\left\langle (\mathbf{y} - \langle \mathbf{y} \rangle) (\mathbf{y} - \langle \mathbf{y} \rangle)^T \right\rangle = \sigma^2 \mathbf{I}_n; \tag{B.15}$$

where σ is the standard deviation of the noise, $\langle \mathbf{y} \rangle$ is the mean value of $\mathbf{y}$ and the expectation is taken over all training sets. Due to the linear characteristic of the model, the relationship between $\mathbf{w}$ and $\mathbf{y}$ is given by

$$\mathbf{w} = \mathbf{G}^{-1} \mathbf{H}^T \mathbf{y}; \tag{B.16}$$

where $\mathbf{G} = \mathbf{H}^T \mathbf{H} + \boldsymbol{\lambda}$. Hence, the expected value of $\mathbf{w}$ is

$$\langle \mathbf{w} \rangle = \langle \mathbf{G}^{-1} \mathbf{H}^T \mathbf{y} \rangle = \mathbf{G}^{-1} \mathbf{H}^T \langle \mathbf{y} \rangle. \tag{B.17}$$

The corresponding variance is given by

$$\begin{aligned} \mathrm{var}(\mathbf{w}) &= \langle (\mathbf{w} - \langle \mathbf{w} \rangle)(\mathbf{w} - \langle \mathbf{w} \rangle)^T \rangle & \text{(B.18)} \\ &= \sigma^2 \mathbf{G}^{-1} \mathbf{H}^T \mathbf{H} \mathbf{G}^{-1}. & \text{(B.19)} \end{aligned}$$

For the case of non-regularization $\mathrm{var}(\mathbf{w}) = \sigma^2 \mathbf{G}^{-1}$. $\mathbf{G}^{-1}$ can be considered as the variance matrix, because of its close link to the variance of $\mathbf{w}$ in the least squares method without regularization. For the global regularization $\mathrm{var}(\mathbf{w}) = \sigma^2 (\mathbf{G}^{-1} - \lambda \mathbf{G}^{-2})$. In the case where the training set is used to estimate the regularization, the matrix $\mathbf{G}$ is a stochastic variable and the relationship between $\mathrm{var}(y)$ and $\mathrm{var}(w)$ is not simply linear any more. This case is not considered in this work.

B.4 SVD scheme for solving linear least squares problems

The sum square errors SSE described in chapters 3, 4, 5 can be rewritten under the short form as follows

$$min \mid \mathbf{G} \mathbf{w} - \mathbf{b} \mid_2, \tag{B.20}$$

where $\mathbf{b}$ is a known vector; $\mathbf{w}$ is the unknown and the design matrix $\mathbf{G} \in \mathbf{R}^{m \times n}, m \geq n$ is generally non-square. Depending on the characteristic of this matrix, the solution of this problem can be either unique or infinitive. If m is less than n or $\mathbf{G}$ is rank deficient, the problem has infinitely many solutions. The single value decomposition (SVD) of a matrix $\mathbf{G}$ is a matrix decomposition of great theoretical and practical importance for the treatment of least squares problems (Bjorck, 1996). Let $r = \min(m, n)$ is the rank of $\mathbf{G}$, the SVD of matrix $\mathbf{G}$ is given by

$$\mathbf{G} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T = \sum_{i=1}^{r} \mathbf{u}_i \sigma_i \mathbf{v}_i^T, \qquad \mathbf{\Sigma} = \begin{pmatrix} \mathbf{\Sigma}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix},$$

where $\Sigma \in \mathbf{R}^{m \times n}$; $\Sigma_1 = \text{diag}(\sigma_1, \sigma_2, ..., \sigma_r)$ with $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_r > 0$, σ_i's are called the singular values of $\mathbf{G}$; $\mathbf{U} = (\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_r)$ and $\mathbf{V} = (\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_r)$.

The condition number of matrix $\mathbf{G}$ influences the perturbation of least squares solutions and is defined as the ratio of the largest and smallest singular values of matrix $\mathbf{G}$ (σ_1/σ_r). The condition number is large then the matrix $\mathbf{G}$ can be ill-conditioned. The problems usually belong to two types of ill conditioning of $\mathbf{G}$: rank deficient and discrete ill-posed problems. While for the former, the numerical treatment is to extract the linearly independent information of $\mathbf{G}$ to form another problem with a well-posed matrix, for the latter, the purpose is to find a balance between the residual norm and the norm of the solution. See Hansen (1998) and Bjorck (1996) for more details.

If $\mathbf{G}$ is ill-conditioned, the minimum variance (B.20) is still large. If the estimator is allowed to be biased, the variance can be drastically reduced. One way to achieve this is to compute the truncated SVD (TSVD) solution in which we assign a numerical rank k that is a positive integer between 1 and r to $\mathbf{G}$. Setting to zero all singular values $\sigma_i, i > k$, the solution of the linear least squares problems can be written as follows

$$\mathbf{w} = \sum_{i=1}^{r} \frac{\mathbf{u}_i \mathbf{b}}{\sigma_i} \mathbf{v}_i, \tag{B.21}$$

which cuts out all the contributions to the solution related to the singular values $\sigma_i, i > k$. Note that the index k is usually referred to as the truncation parameter or the regularization parameter. Hence, the TVSD solution solves the related least squares problem

$$\min_{\mathbf{w}} \left|\mathbf{G}_k \mathbf{w} - \mathbf{b}\right|_2, \qquad \mathbf{G}_k = \sum_{i=1}^{k} \mathbf{u}_i \sigma_i \mathbf{v}_i^T$$

where $\mathbf{G}_k$ is the best rank approximation of $\mathbf{G}$.

Appendix C

Complements to the polymeric kinetic theory and stochastic simulation technique

C.1 Equilibrium configurations

In kinetic theory, one of the ways to determine the initial distribution of the dumbbell configurations $\mathbf{Q}s$ is to use the equilibrium distribution whose probability density is $\psi_e(\mathbf{Q})$. This probability density can be obtained by solving the FPE at equilibrium (i.e. $\mathbf{u} = 0$). The equilibrium distribution is spatial and time independent and given for the Hookean and FENE dumbbell models, respectively, as follows (Bird *et al.*, 1987b).

$$\psi_{e(\text{Hookean})}(\mathbf{Q}) = \frac{1}{(2\pi)^{d/2}} \exp\left(-\frac{|\mathbf{Q}|^2}{2}\right); \tag{C.1}$$

$$\psi_{e(\text{FENE})}(\mathbf{Q}) = \frac{1}{J_e}\left(1 - \frac{|\mathbf{Q}|^2}{b}\right)^{b/2}; \tag{C.2}$$

where (C.1) is nothing but the probability of the Gaussian distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$; d is the dimension of problem; J_e is the normalization factor for the FENE model and given by

$$J_e = \pi b \mathcal{B}\left(1, \frac{b+2}{2}\right) \quad \text{for} \quad d = 2; \tag{C.3}$$

$$J_e = 2\pi b^{3/2} \mathcal{B}\left(3/2, \frac{b+2}{2}\right) \quad \text{for} \quad d = 3; \tag{C.4}$$

where $\mathcal{B}$ is the Beta function and given by

$$\mathcal{B}(u,v) = 2\int_0^1 t^{ux-1}(1-t^2)^{v-1}dt; \tag{C.5}$$

where $u, v \in \mathcal{C}$ and $Re(u) > 0, Re(v) > 0$.

For the Hookean and FENE dumbbell models, the equilibrium stress tensor must be zero ($\bar{\boldsymbol{\tau}}(t, \mathbf{x}) = 0$) and then we have $\langle \mathbf{QF}(\mathbf{Q}) \rangle = \mathbf{I}$, $\forall \mathbf{x} \in \Omega$.

C.2 Zero shear rate viscosity

A typical feature of non-Newtonian fluids is that the viscosity depends on the shear rate and a zero-shear-rate viscosity η_p can be defined for such a fluid. The zero shear rate viscosity is an important parameter for fitting a model to given fluid rheological quantities. The zero shear rate for the dumbbell models can be determined by using the Taylor expansion for the shear stress in the limit of a small shear rate flows at the equilibrium configuration distribution (Bird *et al.*, 1987b).

For the Hookean dumbbell model, since the shear stress is linear in the shear rate, the viscosity is independent of the shear rate and given by

$$\eta_{p(\text{Hookean})} = \lambda_H n k_B T. \tag{C.6}$$

For the FENE dumbbell model, a shear thinning fluid, the viscosity is given by

$$\eta_{p(\text{FENE})} = \frac{\lambda_H n k_B T}{1 + (d+2)/b}, \tag{C.7}$$

where d is the spatial dimension of the problem.

C.3 The Wiener process

The Wiener process $\mathbf{W}$ is a time dependent Gaussian variable. In this process, the process increments $\Delta W_i = W(t_i + \Delta t_i) - W(t_i)$, are uncorrelated

$$\langle \Delta W_i \Delta W_j \rangle = \delta_{ij} \Delta t_i. \tag{C.8}$$

where Δt_i is non-overlapping time intervals. Hence, (C.8) shows the statistical independence of non-overlapping time intervals. From (C.8), since the time increment of a Wiener process scales as $\Delta W \propto \sqrt{\Delta t}$ (i.e. $\langle \Delta W^2 \rangle = \Delta t$), the Wiener process is non differentiable.

C.4 Random number generators

In the microscopic component of the hybrid simulation approach discussed here, the stochastic processes employ random numbers in two stages:

(1.) In guessing the initial values of random variables;

(2.) In discretizing the Wiener process term of the evolution equations.

These random vector variables satisfy either the Gaussian distribution law $\mathcal{N}(0,1)$ (C.1) or some other, for example (C.2). In this section, algorithms for generating the initial configurations of equilibrium Hookean and FENE dumbbells, and the Wiener increments are presented. These algorithms are based on the uniform law $\mathcal{U}(0,1)$ whose random numbers can be obtained from standard scientific computation libraries.

Gaussian random numbers

The initial configuration of Hookean dumbbells at equilibrium and the Brownian (Wiener) increments are generated via the normal law generator which can be found in standard scientific computation libraries. These random numbers are also obtained from either Box-Muller or Polar-Marsaglia methods. Here, only the former is described. More details and the algorithm of the Polar-Marsaglia scheme can be found in Deak (1990) and Kloeden and Platen (1995). In the Box-Muller scheme, the two independent random numbers Q_1, Q_2 satisfying the Gaussian law $\mathcal{N}(0,1)$ are generated as follows

(1.) Generating independently $s_1, s_2 \sim \mathcal{U}(0,1)$;

(2.) $Q_1 = \sqrt{-2\ln(s_1)}\cos(2\pi s_2)$ and $Q_2 = \sqrt{-2\ln(s_1)}\sin 2\pi s_2$.

FENE equilibrium configurations

The method of generating the initial configuration of equilibrium FENE dumbbells is based on Von Newmann's rejection technique (Von Neumann, 1951). Let Q be a desired scalar random number which satisfies a known, bounded probability density $\psi(Q)$, $m \leq Q \leq n$ and $M \geq \psi(Q)$ a positive number. The random number Q is generated according to the following algorithm

(1.) Generating $s_1 \sim \mathcal{U}(m,n)$ as follows

- generating $s_1 \sim \mathcal{U}(0,1)$;
- $s_1 = m + (m-n)s_1$;

(2.) Generating $s_2 \sim \mathcal{U}(0,1)$;

(3.) If $s_2 > \frac{1}{\mathcal{M}}\psi(s_1)$ then goto (1);

(4.) $Q = s_1$,

where $\mathcal{M} = max_{m \leq Q \leq n}\left(\psi(Q)\right)$ is considered to be optimal in the sense that the number of iterations to generate Q is minimal.

For FENE equilibrium distribution (C.2), in order to generate $\mathbf{Q} \in \mathbf{B}_{R^d}(0, \sqrt{b})$, the polar coordinate coordinates $(Q, \alpha) \in [0, \sqrt{b}[\times[0, 2\pi[$ and spherical coordinates $(Q, \alpha, \beta) \in \left[0, \sqrt{b}\right[\times \left[0, 2\pi\right[\times \left[0, \pi\right]$ are employed. $\mathbf{Q}$ then is written for $d = 2$ and 3, respectively as follows

$$\begin{aligned} \mathbf{Q} &= \begin{pmatrix} Q_1 \\ Q_2 \end{pmatrix} = \begin{pmatrix} Q\cos\alpha & Q\sin\alpha \end{pmatrix}^T; \\ \mathbf{Q} &= \begin{pmatrix} Q_1 \\ Q_2 \\ Q_3 \end{pmatrix} = \begin{pmatrix} Q\sin\beta\cos\alpha & Q\sin\beta\sin\alpha & Q\cos\beta \end{pmatrix}^T. \end{aligned} \tag{C.9}$$

Hence, the algorithm for the FENE equilibrium distribution $\mathbf{Q}$ satisfying (C.2) consists of generating the radial coordinate Q and the angular coordinate. $\mathbf{Q}$ is then determined by (C.9). The algorithm is written for 2D and 3D problems as follows

For the 2D problem

(1.) Generating independently $s_1, s_2 \sim \mathcal{U}(0, 1)$;

(2.) If $\left(s_2 > \sqrt{b}\left(1 + \frac{1}{b}\right)^{(b+1)/2} s_1(1 - s_1^2)^{b/2}\right)$ goto (1);

(3.) $Q = s_1\sqrt{b}$;

(4.) Generating $\alpha \sim \mathcal{U}(0, 2\pi)$;

(5.) $Q_1 = Q\cos\alpha$ and $Q_2 = Q\sin\theta$.

For the 3D problem

(1.) Generating independently $s_1, s_2 \sim \mathcal{U}(0, 1)$;

(2.) If $\left(s_2 > \frac{\sqrt{b}}{2}\left(1 + \frac{2}{b}\right)^{(b+2)/2} s_1^2(1 - s_1^2)^{b/2}\right)$ goto (1);

(3.) $q = s_1\sqrt{b}$;

(4.) Generating $\alpha \sim \mathcal{U}(0, 2\pi)$;

(5.) Generating independently $s_3 \sim \mathcal{U}(0, \pi)$ and $s_4 \sim \mathcal{U}(0, 1)$;

(6.) If $(s_4 > \sin s_3)$ goto (**5**);

(7.) $\beta = s_3$;

(8.) $Q_1 = q \sin\beta \cos\alpha$, $Q_2 = q \sin\beta \cos\alpha$ and $Q_3 = q \cos\beta$.

C.5 Computation of connector vectors using predictor-corrector scheme

For illustration, the computation of FENE connector vectors (namely configuration fields) **Q**'s is described and based on eqns (5.13) and (5.14), (similar derivation can be developed for (6.18) and (6.19)), as follows

$$\bar{\mathbf{Q}}_{i+1} = \mathbf{Q}_i + \left[\boldsymbol{\kappa}_i \cdot \mathbf{Q}_i - \frac{\mathbf{Q}_i}{2\left(1 - \frac{\mathbf{Q}_i^2}{b}\right)}\right] \Delta t + \Delta \mathbf{W}_i, \tag{C.10}$$

$$\left(1 + \frac{\Delta t}{4\left(1 - \frac{\mathbf{Q}_{i+1}^2}{b}\right)}\right) \mathbf{Q}_{i+1} =$$
$$\mathbf{Q}_i + \frac{1}{2}\left(\bar{\boldsymbol{\kappa}}_{i+1} \cdot \bar{\mathbf{Q}}_{i+1} + \boldsymbol{\kappa}_i \cdot \mathbf{Q}_i - \frac{\mathbf{Q}_i}{2\left(1 - \frac{\mathbf{Q}_i^2}{b}\right)}\right) \Delta t + \Delta \mathbf{W}_i. \tag{C.11}$$

Eq. (C.11) leads to a cubic equation for Q whose algorithm is described as follows. Let

$$\alpha = |\mathbf{a}| \quad \text{and} \quad Q = |\mathbf{Q}_{i+1}(\mathbf{x})| \tag{C.12}$$

where **a** is the random vector on the RHS of (C.11); $\bar{\mathbf{Q}}_{i+1}$ is calculated by (C.10), $\bar{\boldsymbol{\kappa}}_{i+1}$ is determined by the backward difference scheme (5.15). The cubic equation of Q is given by

$$Q^3 - \alpha Q^2 - b(1 + \frac{\Delta t}{4})Q + \alpha b = 0 \tag{C.13}$$

The intermediate quantities are given by

$$\begin{aligned} L1 &= -\frac{\alpha}{3}\left[\left(\frac{\alpha}{3}\right)^2 - b\left(1 - \frac{\Delta t}{8}\right)\right], \\ L2 &= \left(\frac{\alpha}{3}\right)^2 + \frac{b}{3}\left(1 + \frac{\Delta t}{4}\right), \\ \theta &= \arccos\left(\frac{L1}{L2^{3/2}}\right). \end{aligned}$$

Since the solution Q satisfies $0 \leq Q < \sqrt{b}$, its value is

$$Q = -2\sqrt{Q}\cos\left(\frac{\theta - 2\pi}{3}\right) + \frac{\alpha}{3},$$

and $\mathbf{Q}_{i+1}(\mathbf{x}) = \frac{Q}{\alpha}\mathbf{a}$. More detail can be found in Ottinger, (1996) and Press *et al.* (1992).

References

[1] Adachi, K., Yoshioka, N. and Sakai, K. (1978). An investigation of non-Newtonian flow past a sphere. *J. Non-Newt. Fluid Mech.*, 3, 107-125.

[2] Atluri, S.N., Kim, H.G. and Cho, J.Y. (1999). A critical assessment of the truly mesh-less local Petrov-Galerkin (MLPG) and local boundary integral equation (LBIE) methods. *Comput. Mech.*, 24, 348-372.

[3] Atluri, S.N. and Shen, S. (2002). *The meshless local Petrov-Galerkin (MLPG) method*. Los-Angeles: Tech Science Press.

[4] Babuska, I. and Melenk, J.M. (1997). The partition of unity method. *Int. J. Num. Meth. Engrg.*, 40, 727-758.

[5] Baker, A.J. (1983). *Finite element computational fluid mechanics.* New-York: McGraw-Hill.

[6] Banerjee, P.K. and Butterfield, R. (1979). *Development in boundary element methods-1.* London: Elsevier Applied Science Publisher.

[7] Banerjee, P.K. and Butterfield, R. (1981). *Boundary Element Methods in Engineering Science*. London: McGraw-Hill.

[8] Banerjee, P.K. and Watson, J.O. (1986). *Development in boundary element methods-4*. London: Elsevier Applied Science Publisher.

[9] Barnes, H.A., Hutton, J.F. and Walters, K. (1989). *An introduction to Rheology*. Amsterdam: Elsevier.

[10] Beatson, R.K. and Light, W.A. (1997). Fast evaluation of radial basis functions: methods for two-dimensional polyharmonic splines. *IMA J. Numerical Analysis*, 17, 343-372.

[11] Beatson, R.K., Light, W.A. and Billings, S. (2000). Fast solution of the radial basis function interpolation equations: domain decomposition methods. *SIAM J. Sci. Comput.*, 22(5), 1717-1740.

[12] Becker, S. and Le Cun, Y. (1989). Improving the convergence of back-propagation learning with second order methods, in Touretzky, D., Hinton, G. and Sejnowski, T. (Eds), *Proceeding of the 1988 connectionist models summer school*, 29-37, San Mateo, USA.

[13] Bell, T.W., Nyland, G.H., de Pablo, J.J. and Graham, M.D. (1997). Combined Brownian dynamics and spectral simulation of the recovery of polymeric fluids after shear flow. *Macromolecules*, 30, 1806-1812.

[14] Belli, M.R., Conti, M., Crippa, P. and Turchetti, C. (1999). Artificial neural networks as approximator of stochastic processes, *Neural Networks*, 12, 647-658.

[15] Belytschko, T., Lu, Y.Y. and Gu, L. (1994). Element-free Galerkin methods. *Int. J. Num. Meth. Engrg.*, 37: 229-256.

[16] Belytschko, T., Krongauz, Y., Organ, D., Fleming, M. and Krysl, P. (1996). Meshless methods: An overview and recent developments. *Comput. Methods Appl. Mech. Engrg.*, 139, 3-47.

[17] Beris, A.N., Armstrong, R.C. and Brown, R.A. (1986). Finite element calculation of viscoelastic flow in a Journal bearing: II. Moderate eccentricity. *J. Non-Newt. Fluid Mech.*, 19, 323-347.

[18] Beris, A.N., Armstrong, R.C. and Brown, R.A. (1987). Spectral/finite-element calculations of the flow of a maxwell fluid between eccentric rotating cylinders. *J. Non-Newt. Fluid Mech.*, 22, 129-167.

[19] Beris, A.N., Avgousti, M. and Souvaliotis, A. (1992). Spectral calculations of viscoelastic flows: evaluation of the Giesekus constitutive equation in model flow problems. *J. Non-Newt. Fluid Mech.*, 44, 197-228.

[20] Bernstein, B., Feigl, K.A. and Olsen, E.T. (1994). Steady flows of viscoelastic fluids in axisymmetric abrupt contraction geometry: A comparison of numerical results. *J. Rheol.*, 38(1), 53-71.

[21] Bird, R.B, Armstrong, R.C and Hassager, O. (1987a). *Dynamics of polimeric liquids*, vol 1. New York: John Wiley and Sons.

[22] Bird, R.B, Curtiss C.F., Armstrong, R.C. and Hassager, O. (1987b). *Dynamics of polymeric liquids*, vol 2. New York: John Wiley and Sons.

[23] Bird, R.B, and Wiest, J.M. (1995). Constitutive equations for Polymeric lquids. *Annu. Rev. Fluid Mech.*, 27, 169-193.

[24] Binous, H. and Phillips, R.J. (1999). Dynamic simulation of one and two particles sedimenting in viscoelastic suspensions of FENE dumbbells. *J. Non-Newt. Fluid Mech.*, 83, 93-130.

[25] Bjorck, A. (1996). *Numerical methods for least squares problems.* Philadelphia: SIAM.

[26] Boger, D. V. Crochet, M. J. and Keiller, R. A. (1992). On viscoelastic flows through abrupt contractions. *J. Non-Newt. Fluid Mech.*, 44, 267-279.

[27] Bonvin, J. and Picasso, M., (1999). Variance reduction methods for CONNFFESSIT-like simulations. *J. Non-Newt. Fluid Mech.*, 84, 191-215.

[28] Bonvin, J. (2000). *Numerical simulation of viscoelastic fluids with mesoscopic model*, PhD. thesis, Ecole Polytechnique Federale de Lausanne, Lausanne, Switzerland.

[29] Brebbia, C.A., Telles, J.C.F. and Wrobel, L.C. (1984). *Boundary Element Techniques: theory and applications in engineering.* Berlin: Springer.

[30] Brooks, A.N. and Hughes, T.J.R. (1982). Stream-line Upwind/Petrov-Galerkin formulation for convection dominated flows. *Comput. Meth. Appl. Mech. Engrg.*, 32, 199-259.

[31] Broomhead, D.S. and Lowe, D. (1988). Multivariable functional interpolation and adaptive networks. *Complex Systems*, 2, 321-355.

[32] Carlson, R.E. and Foley, T.A. (1991). The parameter R^2 in multiquadric interpolation. *Computers Math Applic.*, 21(9), 29-42.

[33] Carnahan, B., Luther, H.A. and Wilkes, J.O. (1969). *Applied numerical methods*. New-York: John Wiley and Sons.

[34] Chan, T. and Mathew, T. (1994). Domain decomposition algorithms. *Acta Number.*, 2, 61-143.

[35] Cheng, A.H.D., Golberg, M.A., Kansa, E.J. and Zammito, G. (1985). Some experimental results on the developement of Couette flow for non-Newtonian fluids. *J. Non-Newt. Fluid Mech.*, 17, 125-144.

[36] Chen, C.S., Hon, Y.C. and Schaback, R.A. (2003). *Scientific Computing with radial basis functions*. Draft version 0.0.

[37] Chow, A.W. and Fuller, G.G. (1985). Some experimental results on the developement of Couette flow for non-Newtonian fluids. *J. Non-Newt. Fluid Mech.*, 17, 125-144.

[38] Coleman, C.J, Tullock, D. and Phan-Thien, N. (1991). An effective boundary element method for inhomogeneous partial diferential equations. *ZAMP*, 42, 730-745.

[39] Connor, J.J. and Brebbia, C.A. (1976). *Finite element techniques for fluid flow*. London: Newnes-Butterworths.

[40] Constantinides, A. and Mostoufi, N. (1999). *Numerical methods for chemical engineers with matlab application*. New Jersey: Prentice Hall PTR.

[41] Conti, M. and Turchetti, C. (1994). Approximation of dynamical systems by continuous-time recurrent approximate identity neural networks. *Neural, Parallel and Scientific Computations*, 2, 299-322.

[42] Cormenzana, J., Ledda, A. and Laso, M. (2001). Calculation of free surface flows using the CONNFFESSIT, *J. Rheol.*, 45(1), 237-259.

[43] Cox, D.R. and Miller, H.D. (1977). *The theory of stochastic processes.* New-York: Halsted Press.

[44] Crochet, M.J., Davies, A.R. and Walter, K. (1984). *Numerical simulation of Non-Newtonian flow.* Amsterdam: Elsevier.

[45] Crochet, M.J. and Keunings, R. (1980). Die swell of a Maxwell fluid: Numerical prediction. *J. Non-Newt. Fluid Mech.*, 7, 199-212.

[46] Cybenko, G. (1989). Approximation by superposition of sigmoidal function. *Mathematics of control signals and systems*, 2, 303-314.

[47] Darken, C. and Moody, J. (1990). Fast adaptive k-mean clustering: some empirical results. *Int. Joint Conf. on Neural network*, 2, 233-238

[48] Davies, A.J. and Mushtaq, J. (1996). The domain decomposition boundary element method on a network of transputers. In: Ertekein, R.C., Brebbia, C.A., Tanaka, M. and Shaw, R. (eds). *Boundary Element Technology XI*, 397-406. Southampton: Computational Mechanics Publications

[49] Deak, I. (1990). *Random number generator and simulation.* Budapest: Akademiai Kiado.

[50] Debbaut,B., Marchal, J.M. and Crochet, M.J. (1988). Numerical simulation of highly viscoelastic flows through an abrupt contraction. *J. Non-Newt. Fluid Mech.*, 29, 119-146.

[51] Debae, F., Logat, V. and Crocheta, M. J.(1994). Practical evaluation of four mixed finite element methods for viscoelastic flow. *J. Rheol.*, 38, 421-442.

[52] Demuth, H. and Beale, M. (1998). *Neural network toolbox for use with Matlab*. The MathWorks Inc.

[53] Dissanayake, M.W.M.G. and Phan-Thien, N. (1994). Neural network based approximations for solving partial differential equations. *Communications in Numerical Methods in Engineering*, 10, 195-201.

[54] Doi, M. and Edwards, S.F. (1986). *The theory of polymer dynamics*. Oxford: Clarendon Press.

[55] Dolbow, J. and Belytschko, T. (1999). Numerical integration of Galerkin weak form in meshfree methods. *Computational Mechanics*, 23, 219-230.

[56] Dongarra, J.J., Bunch, J.R., Moler, C.B. and Stewart, G.W. (1979). *PACK user's guide*. Philadelphia: SIAM.

[57] Doob, J.L. (1953). *Stochastic processes*. New York: Johm Wiley and Sons.

[58] Doyle, P.S., Shaqfeh, E.S.G., McKinley, G.H. and Spiegelberg, S.H. (1998). Relaxation of dilute polymer solution following extensional flow. *J. Non-Newt. Fluid Mech.*, 76, 79-110.

[59] Duarte, C.A. and Oden, J.T. (1996). An h-p adaptive method using clouds. *Comput. Methods Appl. Mech. Engrg.*, 139, 237-262.

[60] Dubal, M. R. (1994). Domain decomposition and local refinement for multiquadratic approximations. I: Second order equations in on-demension. *J. applied Science and computation*, 1, 146-171.

[61] Duchon, J. (1976). Interpolation des fonctions de deux variables suivant le principle de la flexion des plaques minces. *RAIRO Analyse Numeriques*, 10, 5-12.

[62] Duda, R.O. and Hart, P.E. (1973). *Pattern classification and scene analysis*. New-York: Wiley.

[63] Dupont, S. and Crochet, M. J. (1988). The vortex growth of a K.B.K.Z. fluid in an abrupt contraction. *J. Non-Newt. Fluid Mech.*, 29, 81-91.

[64] Fan, X.J. (1985). Viscosity, first normal-stress coefficient, and molecular stretching in dilute polymer solutions. *J. Non-Newt. Fluid Mech.*, 17, 125-144.

[65] Fan, X.J., Phan-Thien, N. and Zheng, R. (1999). Simulation of fibre suspension flows by the Brownian configuration field method. *J. Non-Newt. Fluid Mech.*, 84, 257-274.

[66] Fan, X. (1989). Molecular models and flow calculation: II. Simulation of steady planar flow. *Acta Mech. Sin.*, 5:49-59.

[67] Fasshauer, G.E. (1996). Solving partial differential equations by collocation with radial basis functions. In: Le Mehaute, A., Rabut, C. and Schumaker, L. *The proceedings of the 3rd international conference on curves and surfaces, Vol. 2: Surface fitting and multiresolution methods, Chamonix-Mont-Blanc*, 2, 131-138. Tennessee: Vanderbilt University Press.

[68] Fasshauer, G.E. (1999). Solving partial differential equations by collocation with radial basis functions: multilevel methods and smoothing, *Adv. Comp. Math.*. 11, 139-159.

[69] Fasshauer G.E. and Jerome, J.W. (1999). Multistep approximation algorithms: improved convergence rates through postconditioning with smoothing kernels. *Adv. Comp. Math.*, 10, 1-27.

[70] Fedoseyev, A.I., Friedman, M.J. and Kansa, E.J. (2002). Improved multiquadric method for elliptic partial differential equations via PDE collocation on the boundary. *Compt. and Math. Applic.*, 43(3-5), 491-500.

[71] Feigl, K., Laso, M. and Öttinger, H.C. (1995). The CONNFFESSIT approach for solving a two dimensional viscoelastic fluid problem. *Macromolecules*, 28, 3261-3274.

[72] Feigl, K. and Öttinger, H.C. (1996). A numerical study of the flow of a low-density polyethylene melt in a planar contraction and comparison to experiments. *J. Rheol.*, 40, 21-35.

[73] Fine, T.L. (1999). *Feedforward neural network methodology.* Berlin: Springer.

[74] Fixman, M. (1978a). Simulation of polymer dynamics. I. General theory. *J. Chem. Phys.*, 69(4), 1527-1537.

[75] Fixman, M. (1978b). Simulation of polymer dynamics. II. Relaxation rates and dynamic viscosity. *J. Chem. Phys.*, 69(4), 1538-1545.

[76] Fletcher, C.A.J. (1984). *Computational Galerkin methods.* Berlin: Springer-Verlag.

[77] Florez, W.F., Power, H. and Chejne, F. (2000). Numerical solution of non-Newtonian problems by the multi-domain dual reciprocity method. In: Brebbia, C.A. and Power, H. editors, *Boundary Elements XXII*, 613-624, Southampton: WIT Press.

[78] Fornberg, B. Driscoll, T.A. Wright, G. and Charles, R. (2002). Observations on the behavior of radial basis functions near boundaries. *Comput. Math. Appl.*, 43, 473-490.

[79] Franke, R. (1982). Scattered data interpolation: test of some methods, *Math. Comput.*, 48, 181-200.

[80] Franke, C. and Schaback, R. (1998). Solving partial differential equations by collocation using radial basis functions. *Appl. Math. Comput.*, 93, 73-82.

[81] Friedman A. (1975). *Stochastic differential equations and applications.* New York: Academic Press.

[82] Funahashi, F. (1989). On the approximation realisation of continuous mappings by neural networks. *Neural Networks*, 2, 187-192.

[83] Funaro, D., Quarteroni, A. and Zanolli, P. (1988). An iterative procedure with interface relaxation for domain decomposition methods. *SIAM Journal of numerical Analysis*, 25(6), 1213-1236.

[84] Gadalla, H.A., Hua, C.C. and Schieber, J.D. (1996). Nonisothermal Simulation of Polymeric Liquids Using the CONNFFESSIT Approach. In: At-Kadi A., Dealy J.M., James D.F. and Williams M.C. (Eds), *Proceedings of the XIIth International Congress on Rheology, Quebec*, 399-400, Quebec: Laval University,

[85] Gardiner, C.W. (1994). *Handbook of stochastic methods for physics, chemitry and the natural sciences.* Berlin: Springer-Verlag.

[86] Gatski, T.B. and Lumley, J.L. (1978). Steady flow of a non-Newtonian fluid through a contraction. *J. Comp. Phys.*, 27, 42-70.

[87] Gihman, I.I. and Skorohod, A.V. (1974). *The theory of stochastic processes III.* Berlin: Springer-Verlag.

[88] Gingold, R.A. and Monaghan, J.J. (1977). Smoothed particle hydrodynamics: theory and application to non-spherical stars. *Mon. Not. Roy. Astron. Soc.*, 181:375-389.

[89] Girosi, F. and Poggio, T. (1990). Networks and the best approximation property. *Biological Cybernelics*, 63, 169-176.

[90] Girosi, F. (1992). On some extensions of radial basis functions and their applications in artificial intelligence. *Comput. Math. Applic.*, 24(12), 61-80.

[91] Girosi, F., Jones, M. and Poggio, T. (1995). Regularization theory and neural networks architectures. *Neural Computation.*, 7, 219-269.

[92] Golberg, M.A., Chen, C.S. and Karur, S.R. (1996). Improved multiquadric approximation for partial differential equations. *Eng. Anal. with Boundary Elements*, 18, 9-17.

[93] Golberg, M.A. and Chen, C.S. (1997). *Discrete projection methods for integral equations*. Southampton: Computational Mechanics Publication.

[94] Golub, G.H. and van Loan, C.G. (1996). *Matrix computations*. Baltimore: Johns Hopkins University press.

[95] Gottlieb, D. and Orszag, S.A. (1977). *Numerical analysis of spectral methods: Theory and application*. Philadelphia: SIAM.

[96] Griebel, M. and Schweitzer, M.A. (2003). *Meshfree methods for partial differential equations*. Berlin: Springer.

[97] Grillet, A.M., Yang, B., Khomami, B. and Shaqfeh, E.S.G. (1999). Modelling of viscoelastic lid driven cavity flow using finite element simulation. *J. Non-Newt. Fluid Mech.*, 88, 99-131.

[98] Gropp, W., Huss-Lederman, S., Lumsdaine, A., Lusk, E., Saphir, W. and Snir, M. (1998). *MPI-The complete reference, vol 2, the MPI extension*. London: The MIT Press.

[99] Halin, P., Lielens, G., Keuning, R. and Legat, V. (1998). The Lagrangian particle method for macroscopic and micro-macro viscoelastic flow computations. *J. Non-Newt. Fluid Mech.*, 79, 387-403.

[100] Hansen, P.C. (1998). *Rank-deficient and discrete ill-posed problems, Numerical aspects of linear inversion*. Philadelphia: Siam.

[101] Hardy, R.L. (1971). Multiquadric equations for topography and other irregular surfaces. *J. Geophys. Res.*, 176, 1095-1915.

[102] Haykin, S. (1999). *Neural networks: A comprehensive foundation*. New Jersey: Prentice Hall.

[103] He, S., Reif, K. and Unbehauen, R. (2000). Multilayer neural networks for solving a class of partial differential equations. *Neural Networks*, 13, 385-395.

[104] Herrchen, M. and Öttinger, H.C. (1997). A detailed comparison of various FENE dumbbell model. *J. Non-Newt. Fluid Mech.*, 68, 17-42.

[105] Hornik, K., Stinchcombe, M. and White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural networks*, 2, 359-366.

[106] Hua, C.C. and Schieber, J.D. (1998). Viscoelastic flow through fibrous media using the CONNFFESSIT approach. *J. Rheol.*, 42(3), 477-491.

[107] Hughes, T.J.R., Liu, W.K. and Brooks, A. (1979). Finite element analysis of incompressible viscous flows by the penalty function formulation. *J. Comp phys.*, 30, 1-60.

[108] Hulsen, M.A., van-Heel, A.P.G. and van de Brule, B.H.A.A (1997). Simulation of viscoelastic flows using Brownian configuration fields. *J. Non-Newt. Fluid Mech.*, 70, 79-101.

[109] Ingber, M.S. and Phan-Thien, N. (1992). A boundary element approach for parabolic differential equations using a class of particular solutions. *Applied Mathematical Modeling*, 16, 124-132.

[110] Jacobs, R.A. (1988). Increase rate of convergence through learning rate adaptation. *Neural Networks*, 1, 295-307.

[111] Jendrejack, R.M., Graham, M.D. and de Pablo, J.J. (2000). Hydrodynamic interactions in long chain polymer: Application of the Chebyshev polynomial approximation in stochastic simulations. *J. Chem. Phys.*, 113, 2894-2900.

[112] Jourdain, B., Lelivre, T. and Le Bris, C. (2002). Numerical analysis of micro-macro simulations of polymeric fluid flows: A simple case. *Mathematical Models and Methods in Applied Sciences*, 12, 1205-1243.

[113] Kansa, E.J. (1990a). Multiquadrics - A scattered data approximation scheme with applications to computational fluid dynamics-I: Surface approx-

imations and partial derivatives estimates. *Comput. Math. Appl.*. 19(8-9), 127-145.

[114] Kansa, E.J. (1990b). Multiquadrics - A scattered data approximation scheme with applications to computational fluid dynamics-II: Solution to parabolic, hyperbolic and elliptic partial differential equations. *Comput. Math. Appl.*, 19(8-9), 147-161.

[115] Kansa, E.J. and Hon, Y.C. (2000). Circumventing the ill-conditioning problem with multiquadric radial basis functions: Application to elliptic partial differential equations. *Comput. Math. Appl.*, 39, 123-137.

[116] Kao, E.P.C. (1997). *An introduction stochastic process.* Belmont: Duxbury.

[117] Kawahara, M. and Takeuchi, N. (1977). Mixed finite element method for analysis of viscoelastic fluid flow. *Comput. Fluids*, 5, 33-45.

[118] Keunings, R. (1996). On the Petelin approximation for finitely exensible dumbbells. *J. Non-Newt. Fluid Mech.*, 68, 85-100.

[119] Khellaf, K. and Lauriat, G. (2000). Numerical study of heat transfer in a Non-Newtonian Carreau-fluid bwtween rotating concentric vertical cylinders. *J. Non-Newt. Fluid Mech.*, 89, 45-61.

[120] Klebaner F.C. (1998). *Introduction to Stochastic Calculus with Applications.* London: Imperial College.

[121] Kloeden, P.E. and Platen, E. (1995). *Numerical solution of stochastic differential equations.* Berlin: Springer.

[122] Kloeden, P.E., Platen, E. and Schurz, H. (1997). *Numerical solution of SDE through Computer Experiments.* Berlin: Springer.

[123] Kröger, M., Alba-Perez, A. and Öttinger, H.C. (2000). Variance reduced Brownian simulation of a bead-spring chain under steady shear flow considering hydrodynamic interation effects. *J. Chem. Phys.*, 113(11), 4767-4773.

[124] Kubo, R., Toda, M. and Hashitsume, N. (1978). *Statistical physics II.* Springer series in Solid-state sciences 31. Berlin: Springer-Verslag.

[125] Kwan, T.C.B. and Shaqfeh, E.S.G. (1999). Brownian dynamics simulation of the stress and molecular configuration of polymers in exponential and linearly-ramped shear flow. *J. Non-Newt. Fluid Mech.*, 82, 139-165.

[126] Larsson, E. and Fornberg, B. (2001). A numerical study of some radial basis function based solution methods for elliptic PDEs. *Tech. Rep. 470, Dept. of Applied Mathematics*, Univ. of Colorado at Boulder, CO.

[127] Laso, M. and Öttinger, H.C (1993). Calculation of viscoelastic flow using molecular models: the CONNFFESSIT approach. *J. Non-Newt. Fluid Mech.*, 47, 1-20.

[128] Laso, M., Picasso, M. and Öttinger, H.C. (1997). Two dimensional time dependent viscoelastic flow calculations using CONNFFESSIT. *AIchE Journal*, 43(4), 877-892.

[129] Laso, M. (1998). Two-Dimension, time dependent viscoelastic flow calculations using molecular models. In: Adam, M.J., Mashelkar, R.A., Pearson, J.R.A. and Rennie, A.R. (eds), *Dynamics of complex fluids, Proceedings of the second Royal Society Unilever Indo-UK forum in material science and engineering, chapter 6*, 73-87. London: Imperial College Press.

[130] Laso, M., Picasso, M. and Öttinger, H.C. (1999). Calculation of flows with large elongation components: CONNFFESSIT calculation of the flow of a FENE fluid in a planar 10:1 contraction. In: Nguyen, T.Q. and Kausch, H.H., *Flexible polymer chains dynamics in elongational flow: theory and experiment, Chapter 6*, 101-136, Berlin: Springer.

[131] Lawson, C.L. and Hanson, R.J. (1974). *Solving least squares problems.* Englewood Cliffs: Prentice-Hall.

[132] Levenberg, K. (1944). Method for the solution of certain non-linear problems in least squares. *Quart. Appl. Math.*, 2, 164-168.

[133] Le Tallec (1994). Domain decomposition methods in computational mechanics. *J. Comput. Mech. Adv.*, 2, 121-220.

[134] Lielens, G., Keunings, R. and Legat, V. (1999). The FENE-L and FENE-LS closure approximations to the kinetic theory of finitely extensible dumbbells. *J. Non-Newt. Fluid Mech.*, 87, 179-196.

[135] Liu, G.R. (2003). *Mesh-free methods moving beyond the finite element methods*. Boca-Raton: CRC Press.

[136] Liu, W.K., Chen, Y., Uras, R.A. and Chang, C.T. (1996). Generalized multiple scale reproducing kernel particle methods. *Comput. Methods Appl. Mech. Engrg.*, 139, 91-157.

[137] Luo, M.L. and Mitsoulis, E. (1990). A numerical study of the effect of elongational viscosity on vortex grownth in the contraction flows of the polyethylene melts. *J. Rheol.*, 34(3), 309-342.

[138] Macosko, C.W. (1994). *Rheology principles, measurements, and applications*. New-York: Wiley-Vch.

[139] Mai-Duy, N. and Tran-Cong, T. (2001). Numerical solution of Navier-Stokes equations using multiquadric radial basis function networks. *Neural Networks*, 14, 185-199.

[140] Marchal, J.M. and Crochet, M.J. (1987). A new mixed finite element for calculating viscoelastic flow. *J. Non-Newt. Fluid Mech.*, 26, 77-114.

[141] Marini, L.D. and Quarteroni, A. (1988). An iterative procedure for domain decomposition methods: a finite element approach. In R. Glowinski, G.H. Golub, G.A. Meurant and Periaux J., (eds), *Proceedings of first International Symposium on Domain Decomposition methods for partial differential equations*, 129-143, Philadenphia: SIAM.

[142] McKeown, J.J., Stella, F. and Hall, G. (1997). Some numerical aspects

of the training problem for feedforward neural nets. *Neural Networks*, 10(8), 1455-1463.

[143] Melchior, M. and Öttinger, H.C. (1995). Variance reduced simulations of stohastic differential equations. *J. Chem. Phys.*, 103(21), 9506-9509.

[144] Melchior, M. and Öttinger, H.C. (1996). Variance reduced simulations of polymer dynamics. *J. Chem. Phys.*, 105(8), 3316-3331.

[145] Mendelson, M.A., Yeh, P.W., Brown, R.A. and Armstrong, R.C (1982). Approximation error in finite element calculation of viscoelastic fluid flows. *J. Non-Newt. Fluid Mech.*, 32, 197-224.

[146] Mitchell, T.M. (1997). *Machine learning.* New-York: McGraw-Hill.

[147] Mochimaru, Y. (1983). Unsteady-state development of plane Couette flow for viscoelastic fluids. *J. Non-Newt. Fluid Mech.*, 12, 135-152.

[148] Moody, J. and Darken, C.J. (1989). Fast learning in networks of locally-tuned processing units. *Neural Computation*, 1, 281-294.

[149] Nayak, R. (1998). *Molecular simulation of liquid crystal polymer flow: a wavelet-finite element analysis*, PhD. thesis, MIT, Cambridge, MA, USA.

[150] Nayroles, B., Touzot, G. and Villon, P. (1992). Generalizing the finite element method: diffusion approximation and diffuse elements. *Comput. Mech.*, 10, 307-318.

[151] Nguyen-Thien, T., Tran-Cong, T. and Phan-Thien, N. (1997). An improved BEM for analysis of profile polymer extrusion. *Engng. Anal. Boundary Elem.*, 20, 81-89.

[152] Nguyen-Thien, T. (1999). *Some Bie-based numerical methods for continuum mechanics problems*, PhD. thesis, The University of Southern Queensland, Toowoomba, Australia.

[153] Nguyen-Thien, T. and Tran-Cong, T. (2000). BEM-NN approach for polymeric liquids flow analysis. *Engng. Anal. Boundary Elem.*, 24, 95-106.

[154] Novak, A.J. and Brebbia, C.A. (1989). The Multiple Reciprocity Method: A new approach for transforming BEM domain integrals to the boundary. *Engineering Analysis*, 6(3), 164-167.

[155] Onate, E., Idelsohn, S., Zienkiewicz, O.C. and Taylor, L. (1996). A finite point method in computation mechanics. Application to convective transport and fluid flow. *Int. J. Numer. Engng.*, 39, 3839-3866.

[156] Orr, M.J.L. (1997). Matlab routines for subset selection and ridge regression in linear network *http://www.cns.ed.ac.uk/∼mjo.*

[157] Orr M.J.L. (1999a). Local smoothing of radial basis function networks. *http://www.cns.ed.ac.uk/∼mjo.*

[158] Orr M.J.L. (1999b). Regularisation in the selection of radial basis function centres. *http://www.cns.ed.ac.uk/∼mjo.*

[159] Öttinger, H.C. (1994). Variance reduced Brownian dynamics simulation. *Macromolecules*, 27, 3415-3423.

[160] Öttinger, H.C. (1996). *Stochastic processes in polymeric fluids: tools and examples for developing simulation algorithms.* Berlin: Springer.

[161] Öttinger, H.C., van den Brule, B.H.A.A. and Hulsen, M.A. (1997). Brownian configuration fields and variance reduced CONNFFESSIT. *J. Non-Newt. Fluid Mech.*, 70, 255-261.

[162] Pakdel, P., Spiegelberg, S.H. and McKinley, G.H. (1997). Cavity flows of elastic liquids: Two-dimensional flows. *Phys. Fluids*, 9(11), 3123-3140.

[163] Park, J. and Sandberg, I.W. (1991). Universal approximation using radial basis function networks. *Neural Computation*, 3, 246-257.

[164] Park, J. and Sandberg, I.W. (1993). Approximation and radial basis function networks. *Neural Computation*, 5, 305-316.

[165] Partridge, P.W., Brebbia, C.A. and Wrobel, L.C. (1992). *The Dual Reciprocity Boundary Element Method*. Southampton: Computational Mechanics Publication.

[166] Patankar, S.V. (1980). *Numerical heat transfer and fluid flow*. New-York: MacGraw-Hill.

[167] Perera, M.G.M. and Walters, K. (1977). Long-range memory effects in flows involving abrupt changes in geometry. Part I. Flows associated with L-shaped and T-shaped geometries. *J. Non-Newt. Fluid Mech.*, 2, 49-81.

[168] Peyret, R. and Taylor, T.D. (1983). *Computational methods for fluid flow*. Berlin: Springer-Verlag.

[169] Phan-Thien, N. and Tanner, R.I. (1977). A new constitutive equation derived from network theory. *J. Non-Newt. Fluid Mech.*, 2, 353-365.

[170] Phan-Thien, N. (1978). A non-linear network viscoelastic model. *J. Rheol.*, 22, 259-283.

[171] Phan-Thien, N. (1984). Queezing a viscoelastic liquid from a wedge: an exact solution. *J. Non-Newt. Fluid Mech.*, 16, 329-345.

[172] Phan-Thien, N., Tran-Cong, T. and Ramia, M. (1987). A boundary element analysis of Faggellar Propulsion. *J. Non-Newt. Fluid Mech.*, 184, 533-549.

[173] Phan-Thien, N. and Kim, S. (1994). *Microstructures in Elastic Media. Principles and computational methods*. Oxford: Oxford University Press.

[174] Phelan, F.R., Malone, M.F. and Winter, H.H. (1989). A purely hyperbolic model for unsteady viscoelastic flow. *J. Non-Newt. Fluid Mech.* 32, 197-224.

[175] Plawsky, J.L. (2001). *Transport phenomena fundamentals.* New-York: Marcel Dekker.

[176] Poggio, T. and Girosi, F. (1990). Networks for approximation and learning. In: Lau, C. (Ed), *Neural Networks: Theoretical foundations and analysis*, 91-106, New-York: IEEC Press.

[177] Prabhakar, R. and Prakash, J.R. (2002). Viscometric functions for Hookean dumbbells with excluded volume and hydrodynamic interaction. *J. Rheol.*, 46, 1191-1220.

[178] Press, W.H., S.A. Teukolsky, S.A., Vetterling, W.T. and Flannery B.P. (1992). *Numerical recipes in Fortran: the art of scientific computing.* New-York: Cambridge University Press.

[179] Prokhorov, Y.V. and Shiryaev, A.N. (1989). *Probability theory III.* Berlin: Springer.

[180] Quarteroni, A. and Valli, A. (1999). *Domain decomposition methods for partial differential equations.* Oxford: Clarendon Press.

[181] Raghay, S. and Hakim, A. (2001). Numerical simulation of White-Metzner fluid in a 4:1 contraction. *Int. J. Numer. Meth. Fluids*, 35, 559-537.

[182] Randles, P.W. and Libersky L.D. (1996). Smoothed particle hydrodynamics: some recent improvements and applications. *Comput. Methods Appl. Mech. Engrg.*, 139, 375-408.

[183] Reddy, J.N. and Gartling, D.K. (1994). *The Finite Element Method in heat transfer and fluid dynamics.* Boca Raton: CRC Press.

[184] Roache, P.J. (1998). *Fundamentals of computational fluid dynamics.* Albuquerque: Hermosa Publisher.

[185] Rojas R. (1996). *Neural networks: a systematic introduction.* Berlin:Springer.

[186] Ryssel, E. and Brunn, P.O (1999). Comparison of quasi-Newtonian fluid with a viscoelastic fluid in planar contraction flow. *J. Non-Newt. Fluid Mech.*, 86, 309-335.

[187] Sen, A. and Srivastava, M. (1997). *Regression analysis: theory, methods and applications*. Berlin: Springer.

[188] Smith, G.D. (1978). *Numerical solution partial differential equations: Finite Difference Methods*. Oxford: Clarendon.

[189] Smith, B.F., Bjorstad, P.E. and Gropp, W.D. (1996). *Domain decomposition parallel multilevel methods for elliptic partial differential equations*. New-York: Cambridge University Press.

[190] Snir, M., Otto, S., Huss-Lederman, S., Walker, D. and Dongarra, J. (1998a). *MPI-The complete reference*, V1. London: The MIT Press.

[191] Snir, M., Otto, S., Huss-Lederman, S., Walker, D. and Dongarra, J. (1998b). *MPI-The complete reference, the MPI core*, V2. London: The MIT Press.

[192] Soize, C. (1994). *The Fokker-Planck equation for stochastic dynamical systems and its explicit steady state solutions*. Singapore: World Scientific

[193] Somasi, M. and Khomami, B. (2000). Linear stability and dynamics of viscoelastic flows using time-dependent stochastic simulation techniques. *J. Non-Newt. Fluid Mech.*, 93, 339-362.

[194] Suen, J.K.C., Joo, Y.L. and Armstrong, R.C. (2002). Molecular orientation effects in viscoelasticity. In: Lumley J.L., Davis S.H. and Moin P. (eds), *Annual Review of Fluid Mechanics*, 34, 417-444.

[195] Sun, J., Phan-Thien, N. and Tanner, R.I. (1996). An adaptive viscoelastic stress splitting scheme and its applications: AVSS/SI and AVSS/SUPG. *J. Non-Newt. Fluid Mech.*, 65, 75-91.

[196] Tang, W. (1988). *Transforming domain into boundary integrals in BEM.* New-York: Springer.

[197] Talwar, K.K., Ganpule, H.K. and Khomami, B. (1994). A note on selection of spaces in computation of viscoelastic flows using the hp-finite element method. *J. Non-Newt. Fluid Mech.*, 52, 293-307.

[198] Tanner, R.I. (2002). *Engineering rheology.* Oxford: University Press.

[199] Tanner, R.I. and Xue, S.C. (2002). Computing transient flows with high elasticity. *Korea-Australia Rheology Journal.* 14(4), 143-159.

[200] Thomasset, F. (1981). *Implementation of finite element methods for Navier-Stokes equations.* Berlin: Springer.

[201] Tran-Canh, D., Mai-Duy, N and Tran-Cong, T. (2000a). Comparison of BEM-FFNN and BEM-RBFN methods for GNF flow analysis. In Binding, D.M., Hudson, N.E., Mewis, J., Piau J-M., Petrie, C.J.S., Townsend, P., Wagner M.H. and Walters K., (eds), *The XIIIth International Congress on Rheology, Cambridge*, 2, 140-142, Cambridge: The British Society of Rheology

[202] Tran-Canh, D. and Tran-Cong, T. (2000b). Parallel computation of power-law flows using BEM-NN method and subregion techniques on PVM cluster. *The Conference on high performance scientific computing, Hanoi*, 140, Hanoi: The Nation Scientific and Technology Center.

[203] Tran-Canh, D. and Tran-Cong, T. (2002a). BEM-NN computation of Generalised Newtonian Flows. *J. Eng. Anal. with Boundary Elements.* 26, 15-28.

[204] Tran-Canh, D. and Tran-Cong, T. (2002b). Computation of viscoelastic flow using Neural Networks and Stochastic Simulation. *Korea-Australia Rheology Journal*, 14(4), 161-174.

[205] Tran-Canh, D. and Tran-Cong, T. (2003a). Element-free simulation of

dilute polymeric flows using Brownian configuration field. *Korea-Australia Rheology Journal*, accepted.

[206] Tran-Canh, D. and Tran-Cong, T. (2003b). Meshless computation of 2D viscoelastic flows. *Korea-Australia Rheology Journal*, submitted.

[207] Tran-Canh, D. and Tran-Cong, T. (2003c). Parallel meshfree TPS-RBFN-BCF method for dilute polymer solution. *The proceeding, of the 2nd Korean-Australia Rheology Conference, Gyeongju*, 67-70, Gyeongju: The Korean Society of Rheology and The Australian Society of Rheology.

[208] Tran-Cong, T. and Phan-Thien, N. (1988). Three-dimensional study of extrusion processes by boundary element method. I. An implementation of higher order elements and some Newtonian results. *Rheologica Acta*, 27, 21-30.

[209] Tran-Cong, T. (1989). *Boundary element method for some three-dimensional problems in continuum mechanics*, PhD. thesis, The University of Sydney, Sydney, Australia.

[210] Tran-Cong, T., Mai-Duy, N. and Phan-Thien, N. (2002). BEM-RBF approach for viscoelastic flow analysis. *Engineering Analysis with Boundary Element*, 26, 757-762.

[211] Travis, B.J., Anderson, C., Baumgardner, J., Gable, C.W., Hager, B.H., O'connell, R.J., Olson, P., Raefsky, A. and Schubert, G. (1990). A benchmark comparison of numerical methods for infinite Prandrl number thermal convection in two-dimensional Cartesian geometry. *Geophys. Astrophys. Fluid Dynamics*, 55, 137-160.

[212] Trefethen, L.N. and Bau III, D. (1997). *Numerical linear Algebra.* Philadelphia: SIAM.

[213] Van de Brule, B.H.A.A. and Hoogerbrugge, P.J. (1995). Brownian Dynamics simulation of reverse polymeric networks. *J. Non-Newt. Fluid Mech.*, 60, 303-334.

[214] Van den Brule, B., van Heel, T. and Hulsen, M. (1997). Brwonian configuration fields: a new method for simulating viscoelastic fluid flow. *Macromol. Symp.*, 121, 205-217.

[215] Van Kemenade, V. and Deville, M.O. (1994). Application of spectral elements to viscoelastic creeping flows. *J. Non-Newt. Fluid Mech.*, 51,277-308.

[216] Van Kampen, N.G. (1981). *Stochastic process in physics and chemistry.* Amsterdam: North-Holland.

[217] Van-Yee, P. (1998). *Regularized radial basis function networks: theory and applications to probability estimation, classification and time series prediction*, PhD. thesis, McMaster University, Hamilton, Ontario, Canada.

[218] Vapnik, V.N. (1998). *Statistical learning theory.* New-York, Wiley.

[219] Versteeg, H.K. and Malalasekera, W. (1995). *An introduction to computational fluid dynamics.* Harlow: Prentice Hall.

[220] Von Neumann, J. (1951). Various techniques used in connection with random digits. *National Bureau of Standards Applied Mathematics Series*, 12, 36-38.

[221] Wahba, G. and Wold, S. (1975). A completely automatic french curve. *Commun. Statist.*, 4, 1-17.

[222] Wahba, G. (1990). Spline models for observational data. *CBMS-NSF regional conference serie in applied mathematics*, V59, Philadelphia: SIAM.

[223] Wapperom P., Keunings, R. and Legat, V. (2000). The Backward-tracking Lagrangian Particle Method for Transient Viscoelastic flows. *J. Non-Newt. Fluid Mech.*, 91 (2000) 273-295.

[224] Wendland, H. (1995). Piecewise polynomial, positive definite and compactly supported radial basis functions of minimal degree. *Adv. Comput. Math.*, 4, 389-396.

[225] Wendland, H. (1998). Error estimates for interpolation by compactly supported radial basis functions of minimal degree. *J. Approx. Theor.*, 93, 258-396.

[226] Wendland, H. (1999). Meshless Galerkin methods using radial basis function. *Math. Comput.*, 68(228), 1521-1531.

[227] Wu, C.F.J. (1983). On the convergence properties of the EM algorithm. *Annals. of Statistics*, 11, 95-103.

[228] Wu, Z.M. (1992). Hermite-Birkhoff interpolation of scattered data by radial basis functions. *Approx. Theory Appl.*, 8(2), 1-10.

[229] Xu, J. and Zou, J. (1998). Some nonoverlapping domain decomposition methods. *Siam Rev.*, 40(4), 857-914.

[230] Yang, D. (1996). A parallel iterative non-overlapping domain decomposition procedure for elliptic problems. *SIAM Journal of Numerical Analysis*, 16, 75-91.

[231] Zhang, X., Li, X.H., Song, K.Z. and Lu, M.W. (2001). Least-squares collocation meshless method. *Int. J. Numer. Meth. Engng.*, 51, 1089-1100.

[232] Zerroukat, M., Power, H. and Chen, C.S. (1998). A numerical method for heat transfer problems using collocation and radial basis functions. *Int. J. Numer. Meth. Engng.*, 42, 1263-1278.

[233] Zerroukat, M., Djidjeli, K. and Charafi, A. (2000). Explicit and implicit meshless methods for linear advection-diffusion-type partial differential equation. *Int. J. Numer. Meth. Engng.*, 48, 19-35.

[234] Zienkiewicz, O.C. and Taylor, R.L. (2000a). *The Finite Element Method, V1: The basis.* Oxford: Butterworth-Heinemann.

[235] Zienkiewicz, O.C. and Taylor, R.L. (2000b). *The Finite Element Method, V3: Fluid dynamics.* Oxford: Butterworth-Heinemann.

[236] Zheng, R., Phan-Thien, N. and Coleman, C.J. (1991a). A boundary element approach for non-linear boundary value problems. *Comput. Mech.*, 8, 71-86.

[237] Zheng, R., Coleman, C.J. and Phan-Thien, N. (1991b). A boundary element approach for non-homogenous potential problems. *Comput. Mech.*, 7, 279-288.

[238] Zheng, R. and Phan-Thien, N. (1992). Transforming the domain integrals to the boundary using approximation particular solutions: a boundary element approach for non-linear problems. *Appl. Numer. Math.*, 10, 435-445.

[239] Zheng, R. and Phan-Thien, N. (1992). Transforming the domain integrals to the boundary using approximate particular solutions: a boundary element approach for nonlinear problems. *Appl. Numer. Math.*, 10, 435-445.

[240] Zhou, X., Hon, Y.C. and Li, J. (2003). Overlapping domain decomposition method by radial basis functions. *Appl. Numer. Math.*, 44, (1-2), 241-255.