Viscoelastic Mobility Problem Using A Boundary Element Method

Nhan Phan-Thien[1] and Xi-Jun Fan[2]
[1]Division of BioEngineering
The National University of Singapore
Singapore 119260
[2]Institute of High Performance Computing
Singapore, 118261

September 2001

Abstract

In this paper, the complete double layer boundary integral equation formulation for Stokes flows is extended to viscoelastic fluids to solve the mobility problem for a system of particles, where the non-linearity is handled by particular solutions of the Stokes inhomogeneous equation. Some techniques of the meshless method are employed and a point-wise solver is used to solve the viscoelastic constitutive equation. Hence volume meshing is avoided. The method is tested against the numerical solution for a sphere settling in the Odroyd-B fluid and some results on a prolate motion in shear flow of the Oldroyd-B fluid are reported and compared with some theoretical and experimental results.

1 Introduction

In the mobility problem, one wishes to find the rigid body motions of a group of particles in a fluid, given the external forces/torques acting on them, and/or the ambient flow they are subjected to. Mobility problems are central in understanding particle interactions with the surrounding and consequently the evolution of microstructure in a complex fluid. The mobility problem in a viscous fluid is well understood and well documented, e.g., Goldman et al. [2], Kim and Karrila [1].

In a viscoelastic fluid, qualitatively different behaviour in the particle motion has been predicted and indeed observed, for example, Joseph et al. [3]. In the absence of inertia, a sphere settling in a viscous fluid parallel to a plane wall will not deviate from its vertical path. Inertia and weak elasticity (using the second-order fluid model) are predicted to push the particle away from the wall, in contrast to the observation of sphere moving toward the wall at a Deborah number of O(1), Becker et al. [4]. There are indications that three-dimensionality is an important factor, in as much as some qualitatively different behaviour only occurs in three-dimensional flows. In a recent numerical simulation Singh and Joseph [5], using a fictitious domain method and the Oldroyd-B model, were able to show that the particle/wall interaction is indeed a three-dimensional effect, and that the sphere moves toward the wall to a preferred position (which does not occur for two-dimensional flows involving cylinders). This is due to normal (Joseph and Feng [6]), or shear stress contributions (Feng et al. [7]) from viscoelastic effects.

When the particle is slender, it tends to fall with its broad side parallel to gravity when viscoelastic effects dominate, and perpendicular to gravity when inertia dominates (Joseph and Liu [8], Huang et al. [9]). In a simple shear flow and if the fluid is viscous, a slender particle undergoes a periodic motion known as Jeffery's orbit (Jeffery [10]), and this information has been used in constructing useful constitutive equations for fibre suspension (Hinch and Leal [11], Dinh and Armstrong [12], Folgar and Tucker [13], Phan-Thien and Graham [14]). It would be useful to find out how much if indeed viscoelasticity modifies Jeffery's orbit. In order to investigate this, one needs a reliable code to solve the complex three-dimensional mobility problem in a viscoelastic fluid (given the driving force and ambient flow, calculate the rigid-body motion of the particles in a viscoelastic fluid). There are several robust numerical methods based on the finite element techniques that can, in principle, be applied to this problem. However, the massive computation requirement coupled with the need to re-mesh the flow domain at every time step make these methods unattractive at the present time. Joseph and his co-workers have developed a very efficient technique based on a distributed Lagrange multiplier (DLM) and fictitious domain method [15], which they applied successfully to a variety of problems. We found the method indeed efficient, and produced qualitative good results, but may lack the precision required for a quantitative assessment. Concurrent with further developing the DLM method, we also investigate an alterative technique based on an indirect boundary element method (BEM), the Completed Double Layer Boundary Element Method (CDLBEM, Kim and Karrila [1], Phan-Thien and Kim [16]). This is suitably modified for viscoelastic flow calculation. The main attraction of the boundary element method is a reduction in the dimensionality of the problem (only a three-dimensional surface mesh needs be generated), and translating and rotating mesh to a new position can be accomplished without re-meshing. Non-linearities associated with viscoelasticity, modelled by the Oldroyd-B fluid, are handled by the particular solution method [17] using a number of moving points, in the same spirit as the meshless method [18] [19]. In this paper, we report a general method for solving the mobility problem in an Oldroyd-B fluid, but other models could have been chosen. We start with a description of the method, followed by a detailed implementation. Code validation is done with the flow past a sphere, where comparison with axisymmetric results of Tiefenbruck and Leal [20] is made. The results for shear flow past an ellipsoid are then presented.

2 Formulation

The motion of an incompressible viscoelastic fluid is governed by the following equations

$$\nabla\cdot\sigma = \rho\frac{Du}{Dt};\quad \nabla\cdot u = 0;\quad x \in V;\tag{1}$$

where V is the flow domain, ρ is the fluid density, u is the velocity field, and D/Dt is the material time derivative. To these equations, some relevant boundary/initial conditions are imposed. For the mobility problem, one has a finite number (M) of rigid inclusions in the viscoelastic fluid, labelled $n = 1, \ldots, M$; and the external force F^n and torque T^n on a particle n are given,

$$F^n = \int_{S_n} t(y)dS(y);\quad T^n = \int_{S_n} (x - x_c^{(n)}) \times t(y)dS(y);\tag{2}$$

where S_n is its bounding surface, $t = \sigma\cdot n$ is the surface traction, n is the outward unit vector on S_n; and $x_c^{(n)}$ is the mass centre of particle n. The ambient flow may also be prescribed:

$$u(x) = u^{\infty}(x);\quad |x| \to \infty.\tag{3}$$

It is required to find the velocity field everywhere in V; and in particular, the rigid body motion of particle n

$$u(x) = U^n + \omega^n \times (x - x_c^{(n)});\quad x \in S_n;\tag{4}$$

where U^n is the translational and ω^n is the angular velocity of n.

The total stress tensor in a viscoelastic fluid may be arbitrarily decomposed as

$$\sigma = \sigma^N + \tau^v;\tag{5}$$

where σ^N is a Newtonian stress tensor (which is usually, but not necessarily, the "solvent contribution"),

$$\sigma^N = -pI + \mu\left(\nabla u + \nabla u^T\right);\tag{6}$$

with p the hydrostatic pressure , μ a conveniently chosen viscosity value, ∇u the velocity gradient, and τ^v is the "polymer contribution" (the remaining part of the total stress tensor), and is given by a suitable constitutive equations. With this stress splitting, the balance of momentum becomes

$$\nabla\cdot\sigma^N = \rho\frac{du}{dt} - \nabla\cdot\tau^v.\tag{7}$$

2.1 CDLBEM Formulation for Viscoelastic Flow

When inertial and viscoelastic terms are negligible compared to the Newtonian contribution $\nabla\cdot\sigma^N$, Eq. (7) is linear and can be solved by the conventional boundary element method, given some suitable boundary conditions. The advantage of the BEM is that it reduces the dimensionality of the problem by one. For a three-dimensional problem it reduces to a three-dimensional surface computational domain, rather than a full three-dimensional domain, and therefore avoids volume meshing. Re-meshing after every time step is also avoided; the surface mesh can simply be translated and rotated with the particles. When inertial and viscoelastic terms are moderate in size, the terms containing the velocity gradients and stress fields on the right hand side of the momentum equation are usually regarded as known pseudo-body forces in an iteration process, and (7) becomes an inhomogeneous differential equation. The pseudo-body force terms are accounted in the boundary element formulation as volume integrals, which are evaluated based on the velocity and stress field obtained at the previous iteration (Tran-Cong and Phan-Thien [25]). A volume mesh is usually required for this purpose and therefore negates the gain in the reduction of dimensionality.

An alternative is to use a particular solution of the Navier equations to replace the volume integration in the boundary integral formulation (Coleman et al. [17], Zheng et al. [26], Nguyen-Thien [27]). This particular solution can be expressed analytically and volume integration can be avoided. However, we have to be careful that the particular solution used is equal to the volume integral terms, otherwise convergence to the correct solution is not guaranteed. The main principle is that the general solution of the inhomogeneous equation can be treated as the superposition of a general solution of the homogeneous equation and a particular solution of the inhomogeneous equation. The solution of the flow problem is obtained, when the prescribed boundary conditions are satisfied. Hence the solution of Eq. (7) can be decomposed as

$$\sigma^N = \sigma^H + \sigma^P; \tag{8}$$

where σ^H is the solution of the homogeneous equation:

$$\nabla \cdot \sigma^H = 0; \tag{9}$$

and σ^P is a particular solution of the inhomogeneous equation:

$$\nabla \cdot \sigma^P = -\nabla \cdot \tau^v + \rho \frac{Du}{Dt}; \tag{10}$$

where the right hand side is regarded as a known function. The total stress is

$$\sigma = \sigma^H + \sigma^P + \tau^v; \tag{11}$$

and the traction at a point on the surface is

$$t = t^H + t^P + t^v; \tag{12}$$

where $t^H = \sigma^H \cdot n$ and u^H are traction and velocity fields of the homogeneous solution, $t^P = \sigma^P \cdot n$ and u^P are those of the particular solution, and $t^v = \tau^v \cdot n$ is the contribution to the traction from the viscoelastic stress. The velocity field is

$$u = u^H + u^P: \tag{13}$$

Boundary conditions can either be velocity and/or traction boundary conditions; these can be translated into the boundary conditions for the homogeneous solution

$$\begin{aligned} u^H|_S &= u|_S - u^P_S; \\ t^H|_S &= t|_S - t^P|_S - t^v|_S; \end{aligned} \tag{14}$$

where $u|_S$ and $t|_S$ are the prescribed boundary conditions of the flow problems.

Since σ^H satisfies the homogeneous equation (9), the velocity field of the homogeneous solution can be expressed in terms of the double layer density φ (Kim and Karrila [1]),

$$u^H(x) = u^\infty(x) + \int_S K(x;y) \cdot \varphi(y)dS(y); \quad x \in V; \tag{15}$$

where $u^\infty(x)$ is the ambient deformation (i.e., the deformation in the absence of particles), and $K(x;y)$ is the double layer kernel. When x is located on the surface S; the double layer suffers a jump and the boundary integral equation for the homogeneous solution can be written as

$$u^H(x) - u^\infty(x) = \varphi(x) + \int_S K(x;y) \cdot \varphi(y)dS(y); \quad x \in S: \tag{16}$$

Using (13) and the no-slip boundary conditions on S_n leads to

$$U^n + \omega^n \times (x - x_c^{(n)}) - u^P(x) - u^\infty(x) = \varphi(y) + \int_S K(x, y)\cdot\varphi(y)dS(y), \quad x \in S_n. \tag{17}$$

Using the completion process and deflation technique (Kim and Karrila [1]), we obtain the final integral equation that is suitable to numerical implementation for multiparticle system without a container,

$$(1 + H)\varphi = b, \tag{18}$$

where

$$H(\cdot) = K(\cdot) + \sum_{i,k} \varphi^{(k,i)} \Big\langle \varphi^{(k,i)}, \cdot \Big\rangle, \tag{19}$$

where the sum is taken over $i = 1, \ldots, 6$, $k = 1, \ldots, M$, the angular brackets denote the natural product

$$\langle a, b\rangle = \int_S a \cdot b\, dS, \tag{20}$$

$\varphi^{(k,i)}$ is the normalized (with respect to the natural product) eigenvector of K, representing the six rigid body motion modes ($i = 1, 2, 3$: translational, $i = 4, 5, 6$: rotational) of particle $k = 1, \ldots, M$, K is the double layer (integral) operator,

$$K(\cdot) = \sum_k \int_{S_k} K(x, y)\cdot\varphi(y)dS(y), \tag{21}$$

and b is the known vector

$$\begin{aligned} b \;=\; & -u^P - u^\infty - \sum_k \Bigg\{ F_{(k)} + F^p_{(k)} + F^v_{(k)} \\ & - \frac{1}{2}\bigg[(T_{(k)} + T^p_{(k)} + T^v_{(k)}) \times \nabla\bigg] \cdot \frac{G(x, x_c^{(k)})}{8\pi\mu} \Bigg\}. \end{aligned} \tag{22}$$

In (22), $F_{(k)}$ and $T_{(k)}$ are external force and torque acting on particle k, $F^p_{(k)}, F^v_{(k)}, T^p_{(k)}$and $T^v_{(k)}$ are the contributions from the particular solution and the viscoelastic stress to the forces and torques on particle k:

$$\begin{aligned} F^p_{(k)} &= \int_{S_k} t^P(y)dS(y), \quad T^p_{(k)} = \int_{S_k} (x - x_c^{(k)}) \times t^P(y)dS(y), \\ F^v_{(k)} &= \int_{S_k} t^v(y)dS(y), \quad T^v_{(k)} = \int_{S_k} (x - x_c^{(k)}) \times t^v(y)dS(y), \end{aligned} \tag{23}$$

and $G(x, x_c^{(k)})$ is the single layer kernel (Stokeslet):

$$\frac{G(x, x_c^{(k)})}{8\pi\mu} = I + \frac{rr}{r^2}, \quad r = x - x_c^{(k)}. \tag{24}$$

Here we assume the inertial forces and torques on all particles are negligible. The total forces and torques acting on particles consist of the components contributed from the external field, particular solution and viscoelastic stress

$$F^H_{(n)} + F^P_{(n)} + F^v_{(n)} + F_{(n)} = 0, \tag{25}$$

$$T^H_{(n)} + T^P_{(n)} + T^v_{(n)} + T_{(n)} = 0. \tag{26}$$

In the completion process of the solution space, we need to use the force and torque on particle n, $F^{H,ext}_{(n)} = -F^H_{(n)}$ and $T^{H,ext}_{(n)} = -T^H_{(n)}$, for the homogeneous solution on the right hand of Eqs. (22) and (28).

The rigid body motion of particles can be extracted from the solutions of the double layer densities:

$$U_i^{(k)} = -\frac{1}{\sqrt{S^{(k)}}} \Big\langle \varphi^{(k,i)}, \varphi \Big\rangle, \quad \omega_i^{(k)} = -\frac{1}{I_i^{(k)}} \Big\langle \varphi^{(k,i+3)}, \varphi \Big\rangle, \quad i = 1, 2, 3, \tag{27}$$

where $S^{(k)}$ is the surface area of particle k; and $I_i^{(k)}$ represents the surface moment of area of k.

When the double layer densities are known, the velocity at a field point x can then be expressed as

$$u(x) = u^P(x) + u^\infty(x) + \sum_k \int_{S_k} K_{ij}(x;y)\varphi_j(y)dS(y)$$
$$+ \sum_k \left\{ F_{(k)} + F^p_{(k)} + F^v_{(k)} - \frac{1}{2}\left[(T_{(k)} + T^p_{(k)} + T^v_{(k)}) \times \nabla\right] \cdot \frac{G(x;x_c^{(k)})}{8\pi\mu} \right\}. \tag{28}$$

2.2 Particular Solutions

Now the problem reduces to how to obtain the particular solution of Eq. (10). If we denote the right side of this equation to be $-f(x)$, we obtain

$$\sigma^p_{ij,j} + f_i = 0. \tag{29}$$

The problem can be conveniently extended to linear elasticity the particular solution is sought from the Galerkin vector, G, where

$$\nabla^4 G_i + \frac{f_i}{\mu} = 0. \tag{30}$$

It is possible to obtain a analytical solution of Eq. (30) when f_i is a radial basis function (Coleman et al. [17]). Thus we assume that f_i can be approximated by a sum of radial basis functions ψ,

$$-\frac{f_i}{\mu} = \sum_{n=1}^{N} \alpha_{in}\psi\left(\frac{|x - x_n|}{\beta_n}\right), \tag{31}$$

where α_{in} are constants (determined by fitting), and seek a particular solution G_i of the following form

$$G_i = \sum_{n=1}^{N} \alpha_{in}\beta_n^4\phi\left(\frac{|x - x_n|}{\beta_n}\right). \tag{32}$$

With $r = |x - x_n|/\beta_n$, where β_n is a suitably chosen constant for point x_n and usually equal to or larger than the distance to the closest neighbouring point. Introducing $\hat{\phi}(r) = r\phi(r)$, a simple equation for $\hat{\phi}(r)$ is obtained (Zheng et al. [26])

$$\frac{d^4}{dr^4}\hat{\phi}(r) = r\psi(r). \tag{33}$$

For a given $\psi(r)$, it is not difficult to integrate the above equation and obtain a particular solution, $\hat{\phi}(r)$. Then G_i is obtained from Eq. (32); u^p can be determined from the Galerkin vector. The coefficients, α_{in}, are determined by solving a system of linear algebraic equations (31) based on the values of f/μ at N points. The functional form of $\phi(r)$ depends on the radial basis function chosen for $\psi(r)$. Several kinds of basis function have been investigated. We adopt the exponential basis function

$$\psi(r) = \exp(-r^2) \tag{34}$$

because it decays rapidly with r, leading to a quick convergence when solving Eq. (31) for α_n iteratively. With this choice of $\psi(r)$, a particular solution of Eq. (33) is

$$\phi(r) = -\frac{1}{8}\left[\left(r + \frac{1}{2r}\right)\sqrt{\pi}\,\mathrm{erf}(r) + \exp(-r^2) - 2\right], \tag{35}$$

where erf is the error function.

2.3 Oldroyd-B Fluid

One of the popular constitutive equations for dilute polymer solutions is the Oldroyd-B fluid, which is a reasonable model for the Boger fluid in moderate shear rate regime. In this model, the stress tensor can

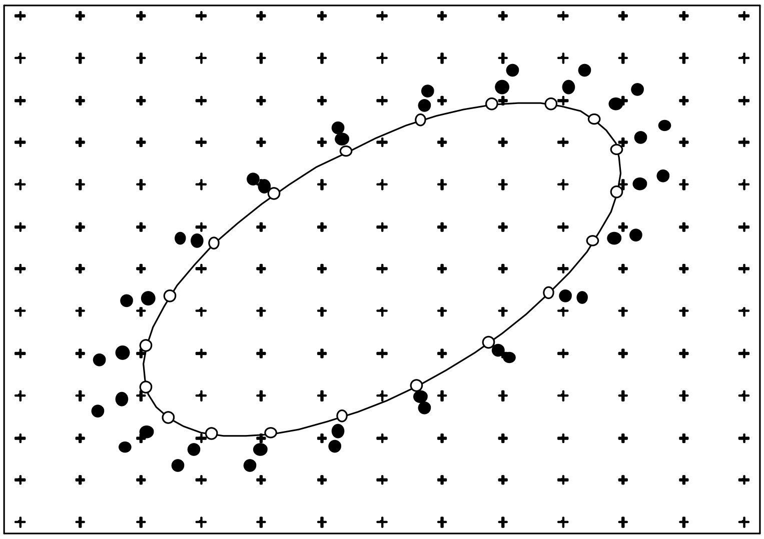

Figure 1: Fixed and moving field points for a 2D case for illustration. Open circles denote the surface element nodes, filled circles denote the moving points and the plus signs denote the fixed points.

be split into two parts: a Newtonian plus a viscoelastic stress, as we did in Eq. (5). The latter can be expressed in the familiar Upper Convected Maxwell (UCM) equation:

$$\tau^v + \lambda\left(\frac{\partial}{\partial t}\tau^v + u\cdot\nabla\tau^v - \nabla u^T\cdot\tau^v - \tau^v\cdot\nabla u\right) = (\eta_r - 1)\mu\dot{\gamma}; \tag{36}$$

where $\eta_r = \eta/\mu$ is the relative viscosity of polymer solutions, λ is the relaxation time of the fluid and $\dot{\gamma} = \nabla u + \nabla u^T$ is twice the strain rate tensor. Introducing the configuration tensor C

$$\tau^v = (\eta_r - 1)\frac{1}{\lambda}(C - I); \tag{37}$$

and substituting it into the constitutive equation for the Oldroyd-B fluid, we have

$$C + \lambda\left(\frac{d}{dt}C - \nabla u^T\cdot C - C\cdot\nabla u\right) = I: \tag{38}$$

In the present simulation, we solve the Eq. (38) for the configuration tensor and then obtain viscoelastic stress through Eq. (37).

3 Numerical Methods

3.1 Field points

One of the important features of the present method is to avoid volume meshing in solving the constitutive equation and obtaining the particular solution in the flow domain. This makes the method flexible in dealing with problems with complex moving boundaries. Similar to the meshless methods (Belytschko et al. [18]; Duarte and Oden [19]; Oñate et al. [28]), we use distributed points in the computational domain instead of volume meshing to do numerical interpolation, differentiation and to solve differential equations. It is obvious that if the number of points is large, and if they are distributed evenly in the field, the solution would be more accurate and stable. However, the computation time would increase with the number of points. We are dealing with infinite flow domain problems, and it would require an infinite number of field points to be distributed in whole domain. Fortunately, for the boundary integral equations, what we need to know is just the values of the velocities and tractions on the boundary. We would like to reduce the number of points as many as possible, provided the solution is stable and accurate enough for the problems in hand. Hence we may distribute more points in the area relatively close to the boundaries to capture potentially large velocity and stress gradients there.

The particles will move along some trajectories during the simulation. We thus use a coordinate system which is located at and moves with the mass centre of the particle system and classify field points into two categories: the points fixed relatively to the coordinate system and the points moving and rotating with each particle. We call the former the fixed points and the latter the moving points. The field points in a 2D case for an elliptical particle are sketched in Fig. 1. The moving points are represented by filled circles, which are distributed in a thin layer around the surface of particle. The fixed points are represented by the plus signs and distributed in a larger area including inside the particle. The fixed points can be either regularly or randomly distributed. From the figure, it is seen that some points are located inside the particle. Hence we have to detect which points are inside the particle and disable them after moving the particle to a new position at each time step. To detect if a point is inside of an ellipsoids, we employed Perram's contact function [29]. Some fixed points, either coincident with, or too close to moving points, are all disabled in this process.

3.2 Fixed Least Square Method

The moving least square method (Lancaster and Salkauskas [30]) has been widely used in meshless methods, such as the element free Galerkin method (Belytschko et al. [18]), the reproducing kernel particle method (Swegle et al. [31]), the $h-p$ clouds (Duarte and Oden [19]), and the finite point method (Oñate et al. [28]), to name a few. This is a locally fitted technique based on randomly distributed points. A similar technique called the fixed least square is employed in the present method. The advantage of the fixed least square method is its simplicity in calculating derivatives but its results are more sensitive to the support chosen than those of the moving least square method. However, the support dependence is not a serious disadvantage here. This is explained below.

First, we outline the fixed least square method. A local approximation of function u at $x \subset \Omega_y$ can be defined as

$$L_y u(\xi) = \sum_{i=1} a_i(y) P_i(\xi) \tag{39}$$

where Ω_y is the compact support of y; P_i is the i-th basis function, and ξ is the local coordinates at y:

$$\xi = [(x_1 - y_1)/h_1, (x_2 - y_2)/h_2, (x_3 - y_3)/h_3], \tag{40}$$

with h_i the size of the support. We usually use monomials $\left\{ P_i(\xi) \right\}_{i=1}^{K}$ as the basis functions. In three-dimensional space (x_1, x_2, x_3), they can be expressed in the local coordinates of y as

$$\begin{cases} \{1\}, & k = 0; \\ \left\{1, \xi_1, \xi_2, \xi_3\right\}, & k = 1; \\ \left\{1, \xi_1, \xi_2, \xi_3, \xi_1^2, \xi_1\xi_2, \xi_2^2, \xi_2\xi_3, \xi_3^2, \xi_3\xi_1\right\}, & k = 2; \\ \ldots & \ldots; \end{cases} \tag{41}$$

where k is the order of the monomials. If $\left\{ P_i(\xi) \right\}_{i=1}^{K}$ are linearly independent over the given N ($N \geq K$) points, $x_I \in \Omega_y$; the coefficients, $a_i(y)$, are determined by the least square method, i.e., to find $a_i^*(y)$ such that

$$J(a^*) = \left(u(x_I) - \sum_{i=1} a_i^*(y) P_i(\xi_I), u(x_I) - \sum_{i=1} a_i^*(y) P_i(\xi_I) \right)_y \leq J(a), \tag{42}$$

where ξ_I is the position vector of point x_I in the local coordinates, and $(\cdot, \cdot)_y$ is a weighted inner product in the support of y, Ω_y:

$$(u, v)_y = \sum_{I} u(\xi_I) W_I(y) v(\xi_I), \tag{43}$$

where $W(y)$ is a weighted function, $W(y) > 0$ when $x \in \Omega_y$, otherwise $W(y) = 0$; and $W_I(y)$ is the value of $W(y)$ at x_I. Solving the inequality (42) requires the derivatives of $J(a)$ with respect to $a_j(y)$ to be zero when $a_j = a_j^*$, i.e.,

$$\sum_{i=1} (P_i, P_j)_y \, a_i^* = (u, P_j)_y, \quad j = 1, \ldots, K. \tag{44}$$

The coefficients a_i^* can be determined from the above equations. If we denote $A_{ij}(y) = (P_i, P_j)_y$ and $B_{jl}(y) = W_l(y) P_j(\xi_l)$ and note the symmetry in A, $A_{ij}(y) = A_{ji}(y)$, this equation can be rewritten as

$$\sum_{j=1} A_{ij}(y) a_j(y) = \sum_{l=1} B_{il}(y) u(x_l). \tag{45}$$

Hence, the solution is

$$a_i(y) = \sum_{l=1} \sum_{j=1} A_{ij}^{-1} B_{jl}(y) u(x_l). \tag{46}$$

Here the superscript $*$ on a_i has been omitted. Substituting $a_i(y)$ into Eq.(39), we obtain the local approximation of $u(x)$ to be

$$L_y u(\xi) = \sum_{l=1} \phi_l^y(\xi) u(x_l), \tag{47}$$

where $\phi_l^y(\xi)$ is the shape function defined as

$$\phi_l^y(\xi) = \sum_{i=1} \sum_{j=1} P_i(\xi) A_{ij}^{-1} B_{jl}(y). \tag{48}$$

Substituting Eq. (40) into Eq. (47), we obtain the locally interpolated values of the function u at x in Ω_y based its values at points x_l, $l = 1, \ldots, N$. The derivatives of local approximation of u with respect to x_m is easy to calculate, since only $P_i(\xi)$ depends on x_m in Eqs (47) and (48),

$$\frac{\partial}{\partial x_m} \phi_l^y(\xi) = \frac{1}{h_m} \sum_{i=1} \sum_{j=1} A_{ij}^{-1} B_{jl}(y) \frac{\partial}{\partial \xi_m} P_i(\xi); \tag{49}$$

and

$$\frac{\partial}{\partial x_m} L_y u(\xi) = \sum_{l=1} \frac{\partial}{\partial x_m} \phi_l^y(\xi) u(x_l). \tag{50}$$

As x approaches to y, we obtain the derivatives of the local approximation of u at y:

$$\frac{\partial}{\partial x_m} L_y u(0) = \sum_{l=1} \frac{\partial}{\partial x_m} \phi_l^y(0) u(x_l). \tag{51}$$

The weighting function we used is

$$W_l(y) = \begin{cases} \left(\frac{4}{\pi}\right)^{3/2} \left(1 - \frac{(y_1 - x_{l1})^2}{h_1}\right)^4 \left(1 - \frac{(y_2 - x_{l2})^2}{h_2}\right)^4 \left(1 - \frac{(y_3 - x_{l3})^2}{h_3}\right)^4, & \text{for } |y_j - x_{lj}| < h_j \\ 0, & \text{otherwise.} \end{cases} \tag{52}$$

It can be seen that the weighting function for the fixed least square method we used here is constant and hence it does not require to calculate the derivative of the weighting function and A^{-1}, when calculating the derivatives of local approximation of a function. This is an advantage of the fixed least square method over the moving least square method and multi-fixed least square method. However, the fixed least square method has its disadvantage as well. It defines the local approximation of a function in each support. When the supports belong to different points overlap and one point may belong to more than one supports, the interpolation and differentiation are multivalued according to the choice of the support. The decision has to be made to limit the choice of the support. In the present method, we only used the fixed least square method to interpolate the function to points which are very near to the data points, and to calculate the derivatives just right at the data points. It is easy to choose a right support which gives the best fitting. The situation here is quite different from that of the element-free Galerkin method and other meshless method to solve differential equations.

During the simulation, it is frequently required to determine a support and create a list of all points within the support for each point. A support should contain enough points, at least larger than K, in order to guarantee the invertibility of the matrix A in Eq. (46). In practice, the number of points within a support is usually much larger than K. When the matrix A is found singular, the support should be enlarged to contain more points. Hence, a efficient search algorithm requiring minimum computation efforts is important. The algorithm reported by Swegle et al. [31] has been employed in the present method. This algorithm consists of three steps: sort, search and compare. The execution time of the algorithm is of $O(N \log_2 N)$ for N points and N search regions.

3.3 Point-wise Solver for Constitutive Equations

The stress tensor for the Oldroyd-B fluid has been expressed by Eqs. (37) and (38) in a previous section. The velocity field is known after the solution of boundary integral equations in every time step. If Eq. (38) can be solved based the known kinematics, the stress tensor would be obtained from the configuration tensor C. The key problem in solving this equation is how to deal with the time derivative of C.

If the time derivative of C is treated in the Eulerian sense, Eq. (38) is not always solvable. For example, if a field point, x, is inside a particle at time $t^{(n-1)}$ but is in the fluid at time $t^{(n)}$ due to the particle motion, to solve for $C(x;t^{(n)})$ we need to know $C(x;t^{(n-1)})$ and its gradient but they cannot be determined since x was disabled at time $t^{(n-1)}$. Alternatively, if the time derivative of C in Eq. (38) is treated in the Lagrangian sense, an implicit finite difference form of this equation can be written as

$$\left(1+\frac{\Delta t}{\lambda}\right) C(X,t^{(n)}) - \Delta t\left[\nabla u^T \cdot C(X,t^{(n)}) + C(X,t^{(n)})\cdot \nabla u\right] = C(X,t^{(n-1)}) + \frac{\Delta t}{\lambda} I, \qquad (53)$$

where $X = X(x,t^{(n)})$ denotes a fluid particle which occupies the field point x at time $t^{(n)}$, and $C(X,t^{(n-1)})$ is the value of the configuration tensor of this fluid particle at time $t^{(n-1)}$. To determine $C(X,t^{(n-1)})$, one usually has to trace the fluid particle backward to find its position at time $t^{(n-1)}$, $x' = x'(X,t^{(n-1)})$, and get the value of the configuration tensor at this position. Here we suppose that the configuration tensor at all field points has been solved at time $t^{(n-1)}$ and the fluid particles occupied these field points will move to new positions at time $t^{(n)}$ carrying the values of the configuration tensor determined at time $t^{(n-1)}$. Though these new positions do not coincide with the field points at time $t^{(n)}$, the convection field of the configuration tensor of $t^{(n-1)}$ has been known and the values at the field points, $C(X,t^{(n-1)})$, can be interpolated from those at the new positions.

As mentioned in section 3.1, the coordinate system moves with the mass centre of the particle system. Hence, the translation of the coordinate system has to be taken into account. Assume that the position of a field point i in the coordinate system at time $t^{(n-1)}$ is y', and the velocity of the mass centre of the particle system at $t^{(n-1)}$ is u_0. At time $t^{(n)}$, the displacement of the coordinate system is $u_0\Delta t$,where $\Delta t = t^{(n)} - t^{(n-1)}$. The position of this field point relative to the coordinate system at $t^{(n)}$ should be $y' - u_0\Delta t$. If the fluid particle that occupied position y'at $t^{(n-1)}$ has velocity $u(y')$, it would move to $y = y' + (u(y') - u_0)\Delta t$ at time $t^{(n)}$ and carry the value of the configuration tensor at $t^{(n-1)}$ to y. Hence we know the convection field of the configuration tensor of time $t^{(n-1)}$, i.e., $C\left(y,t^{(n-1)}\right)$, and from which we can obtain $C\left(x,t^{(n-1)}\right)$ in terms of the interpolation of the fixed least square method, where x is the position of a field point at which the fluid particle X occupies at time $t^{(n)}$.

When $C(X,t^{(n-1)})$ is obtained, $C(X,t^{(n)})$ can be determined by solving Eq. (53). Since C is a symmetrical tensor of the second order, it has six independent components. We only need to solve a linear equation system with 6 unknowns point by point based on the kinematics obtained at previous time step. The initial condition for the configuration tensor can be set optionally, for example $C(X,0) = I$.

3.4 Numerical procedures

Initially, the configuration tensor is set to be the unit tensor, i.e., zero viscoelastic stress tensor. Hence, the velocity of particular solutions, u^p, and the force and torque on particles due to the particular solution and viscoelastic stress, $F^p_{(n)}$, $F^v_{(n)}$, $T^p_{(n)}$, $T^v_{(n)}$, in Eq. (18) are all equal to zero. The ambient flow field, external forces and torques acting on particles are given. The initial positions of particles and moving points are known. The origin of coordinates are located at the mass centre of particles and the positions of the fixed points are fixed with the coordinates. All fixed points which are covered by any particle and too close to any moving points or element nodes are disabled. At each time step, the following procedures are conducted:

1. Solve the boundary integral equation, Eq. (18) for the double layer densities using the boundary element method;
2. Extract the rigid body motion of particles from the double layer densities using Eqs. (27);
3. Move the particles and moving points to the new positions according to their velocities determined in step 2.
4. Calculate fluid velocities at each active fixed points and moving points using Eq. (28);
5. Calculate the convection field of $C(X; t^{(n-1)})$ based the configuration tensor field at the previous time step, using interpolation of the fixed least square method;
6. Check all fixed points to find points which are covered by particles or too close to the element nodes or moving points and disable them;
7. Create a support for each active fixed points, moving points and element nodes (for simplicity, these points are called active points thereafter) and a list of points and nodes in each support;
8. Calculate the velocity gradient at each active points using the fixed least square method;
9. Solve the evolution equation for the configuration tensor, $C(X; t^{(n)})$, Eq. (53) and obtain the viscoelastic stress tensor, Eq. (37), and its divergence using the fixed least square method at each active points;
10. Calculate the pseudo-body force at each active points, i.e., f_i in the left hand side of Eq. (31) and solve this equation for α_{in} using GMRES algorithm, and calculate the velocity and stress due to the particular solution, u^p and σ, according to Eqs. (??) and (??);
11. Calculate the force and torque due to the particular solution and viscoelastic stress acting on the particles and go to step 1. until the final time step is reached.

    All simulations were carried out in the cluster of Compaq Alpha workstations using Parallel Virtual Machine (PVM) library software.

4 Numerical Examples

4.1 Sphere Falling in an Oldroyd-B Fluid

The indirect CDL-BEM formulation is then applied to the simulation of a sphere sedimenting under gravity in the Oldroyd-B fluid for verification. The radius of sphere is normalized to unity, and $294, 384$ and 486 surface elements are used. The fluid is quiescent and fills an infinite space. To minimize computation cost we use as small number of field points as possible and finally 9052 and 25785 fixed points are used. The field points are not uniformly generated. To generate the former configuration, we firstly distribute points evenly in three subregions as follows: (a) $18 \times 18 \times 18$ points in the domain $0 \le |x_1|, |x_2|, |x_3| \le 1.395$, (b) $14 \times 14 \times 14$ points in $0 \le |x_1|, |x_2|, |x_3| \le 2.25$ and (c) $11 \times 11 \times 11$ points in $0 \le |x_1|, |x_2|, |x_3| \le 3.0.$ Then all points located in the domain where denser configuration of points had been generated are deleted. The latter configuration is generated in such a way that $25 \times 25 \times 25$ point are evenly distributed in $0 \le |x_1|, |x_2|, |x_3| \le 1.90$, $21 \times 21 \times 21$ points in $0 \le |x_1|, |x_2|, |x_3| \le 3.0$ and $16 \times 16 \times 160$ points in $0 \le |x_1|, |x_2|, |x_3| \le 5.0.$ Four layers of moving points are distributed in a thin layer on the sphere surface. They are located on the line through the centre of the sphere and the node of a surface element with the distance of $1.015, 1.05, 1.10$ and 1.20 of sphere's radius from the centre. Hence the total number of moving points is equal to $4\times$ number of surface elements.

The present method assumes negligible particle inertia. The initial condition for the configuration tensor was $C(X; 0) = I$, i.e., the initial viscoelastic stress was zero. Hence the sphere was settling as it

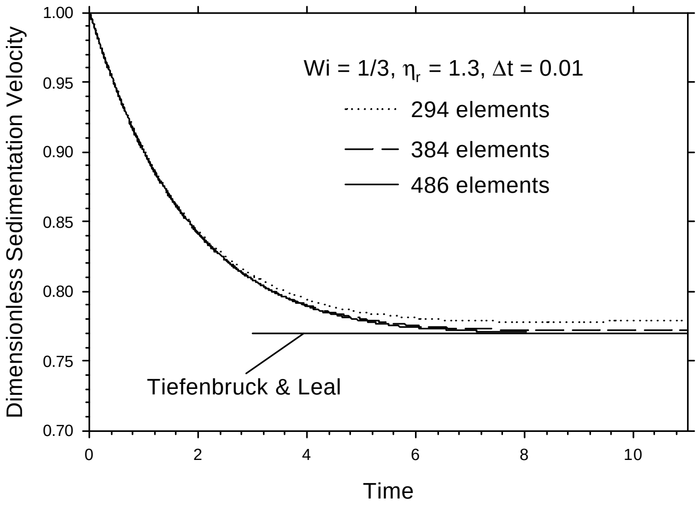

Figure 2: Dimensionless settling velocity of a sphere for ¸ = 1:94932, ´$_r$ = 1:3; ¢t = 0:01 with various surface meshes and 9052 field points.

were in Newtonian fluid initially, and its sedimentation velocity would slow down gradually as viscoelastic stresses were building up and finally, its settling velocity would reach a steady value.

Tiefenbruck and Leal [20] reported an axisymmetric numerical method for streaming flow past a rigid sphere and a spherical bubble in an Oldroyd-type fluid. Their results for the Oldroyd-B fluid can be directly compared with those of the present method. They reported that the dimensionless drag force on the sphere were 2:999; 2:997; 2:985 and 2:98 at Weissenberg number of 0:1; 1=3; 2=3 and 1:0, respectively. The dimensionless drag force is defined as

$$f^{¤} = \frac{drag}{2¼¹´_r Ua}; \qquad (54)$$

where U corresponds to the steady settling velocity and a is the radius of the sphere. The Weissenberg number is defined as ¸U=a: In the present simulation, we set ´$_r$ = 1:3; ¹ = 1:0 and gravity force on the sphere was $\frac{4¼}{3}$ and balanced with the drag force. The steady settling velocity should be

$$U = \frac{2}{3:9f^{¤}}: \qquad (55)$$

The dimensionless settling velocities are plotted in Figs.2, 3 and 4 for ´$_r$ = 1:3 and ¸ = 1:94932; 3:89064

and 5:84795, where the dimensionless sedimentation velocity is defined as the ratio of the sedimentation velocity of the sphere in the Oldroyd-B fluid to that in Newtonian fluid, which is known from the solution of a Stokes flow past a sphere and is equal to 2=9 under the condition mentioned above. In these simulations, 9052 field points and 294, 384 and 486 surface elements were used. Tiefenbruck and Leal's results are plotted in the figures as well, for comparison. From these figures, we can see that the settling velocity calculated by the present method approaches to steady-state solutions of Tiefenbruck and Leal as the viscoelastic force develops and that the difference between two set of results decreases as the number of surface elements increases for ¸ = 1:94932 and 3:89864; i.e., the Weissenburg number to be about 1=3 and 2=3. However, Fig. 4 shows that for ¸ = 5:84795, i.e., Wi t 1:0, the solutions of the present method are divergent due to the accumulation of numerical errors. The numerical error is mainly due to insufficient number of field points. In the above mentioned simulations, the field points were only distributed within a domain of about 6 times the size of the sphere. The minimum separation between points is about 0:164 radius of the sphere. This domain is too small to cover the disturbed flow region by the sphere, especially

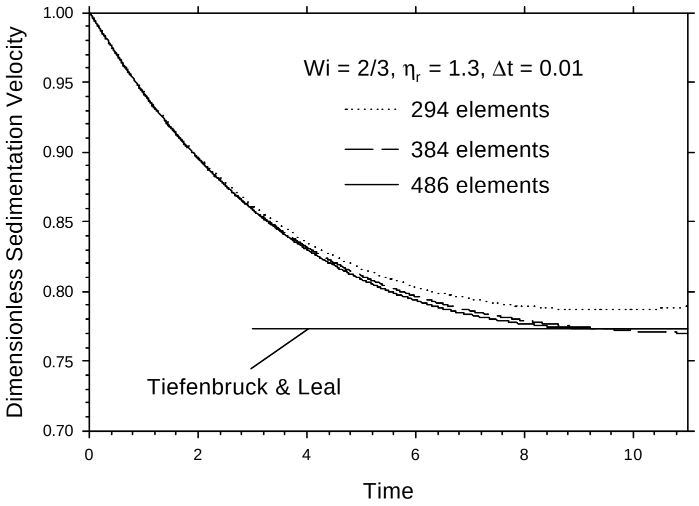

Figure 3: Dimensionless settling velocity of a sphere for ¸ = 3:89864, ´$_r$ = 1:3, ¢t = 0:01 with various surface meshes and 9052 field points.

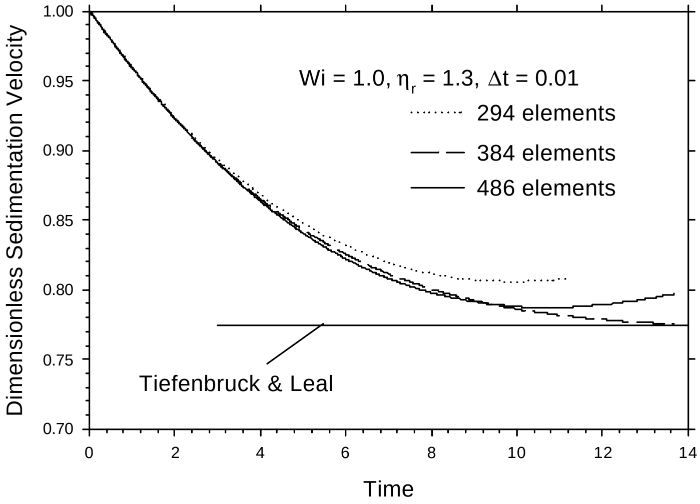

Figure 4: Dimensionless settling velocity of a sphere for ¸ = 5:84795, ´$_r$ = 1:3, ¢t = 0:01 with various surface meshes and 9052 field points.

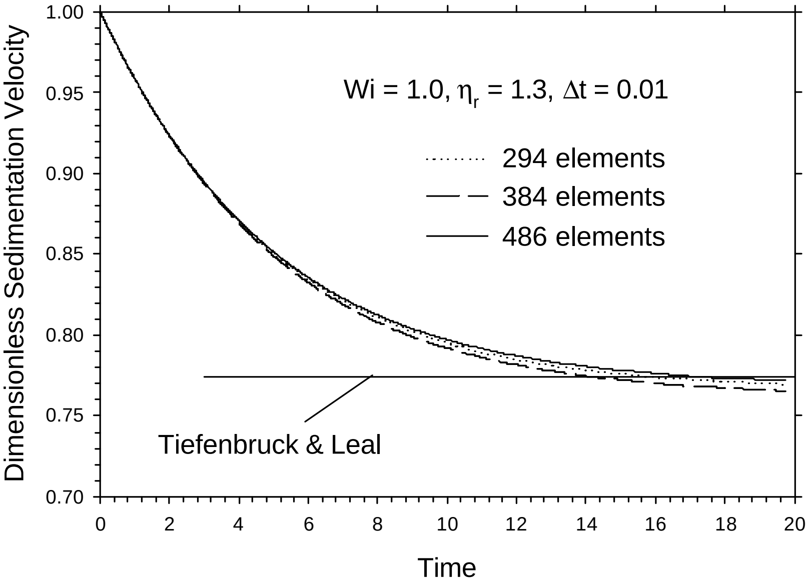

Figure 5: Dimensionless settling velocity of a sphere for $\lambda = 5.84795$, $\eta_r = 1.3$, $\Delta t = 0.01$ with various surface meshes and 25785 field points.

as large Weissenberg number. We next distributed the field points into a lager domain of which the size was about 10 times of the radius and generated the second configuration with 25785 field points and the minimum separation between points to be 0.158 radius of the sphere. The dimensionless settling velocities of the sphere for $Wi = 1.0$ using 25785 field points are shown in Fig. 5. Comparing with Fig. 4, we can see the results are improving significantly: the solution is convergent and the error relative to Tiefenbruck and Leal's is small, 0.35% for 486 surface elements, 1.16% for 384 surface elements and 0.63% for 296 surface elements. However, the computation time increases drastically with the number of field points. This places a constraint on the practical number of field points.

In all above simulations, the time step is chosen to be 0.01. The explicit time dependence is contained in Eq. (53), from which we can see that the solution depends only on the parameter $\Delta t/\lambda$, i.e. Δt can be increasing with λ. However, the error in determining particle's configuration would increase with Δt and the accuracy of the solution would be doubtful if Δt is too large. Hence, Δt is still required to be small enough even for large λ. We found that $\Delta t = 0.01$ is suitable for $\lambda = O(1)$. For smaller Δt, such as 0.005; or larger one, such as 0.02; the final settling velocity changes by about 1~2%. This is due to the accumulation of numerical errors using 9052 field points. This percentage "error" increases with λ but decreases with the number of field points, at a given Δt.

4.2 Prolate spheroid in shear flow

In a Newtonian fluid, a force- and torque-free prolate in shear flow rotates along a Jeffery's orbit, which is an analytical Stokes solution. However, if the fluid is non-Newtonian, the orbit would deviate from Jeffery's due to the viscoelasticity effect. Though there are no complete analytical solution available for a prolate spheroid in a viscoelastic shear flow so far, some of analyses on fibre's motion in shear flow have been reported in the literature. Leal [21] obtained a asymptotic solution of a rod-like particle moving in shear flow of a second-order fluid at low Weissenberg number. He showed that the second-normal-stress difference of the fluid causes a drift across Jeffery orbits towards the vorticity axis. Harlen and Koch [23] analyzed fibres in shear flow of dilute Hookean dumbbell solutions at high Weissenburg number and found the similar spiral motion of fibres but independent of the second-normal-stress difference. Both analyses were based on the assumption that the elastic stress is much smaller than the viscous stress and suggested that at low Weissenberg number the motion of fibres spiral towards vorticity axis depends on a small parameter, which is proportional to the second-normal- stress difference. At high Weissenberg number it can be characterized by one parameter, β, which is independent of the second-normal-stress difference [24]:

$$\beta = \frac{\eta_r - 1}{Wi}. \tag{56}$$

For sufficient small β; the fibre follows approximately Jeffery's orbits but slowly crosses to orbits with progressively lower Jeffery orbital constants and thereby spiral toward the vorticity axis. The period of the motion, T, increases with β in such a way that

$$T = \frac{2\pi(a_r + a_r^{-1})}{\dot{\gamma}(1 - \beta^2 a_r^2/4)^{\frac{1}{2}}}. \tag{57}$$

At the critical value, $\beta_c = 2/a_r$, the fibre remains to the flow-vorticity plane and no longer rotates in Jeffery orbits. Then the fibre will slowly rotate within the flow-vorticity plane until it alines with the vorticity axis. When $\beta = 0$, Eq. (57) predicts the period of the Jeffery orbits, T_0, for Newtonian flow. We can further express the ratio of the non-Newtonian period to the Newtonian one as a function of β/β_c:

$$\frac{T}{T_0} = \left[1 - (\beta/\beta_c)^2\right]^{-\frac{1}{2}}. \tag{58}$$

The increase of the period was confirmed by observations of Bartram et al [22], who found that the period of rotation of a particle in non-Newtonian fluid is considerable longer than that in Newtonian fluid. Iso et al [24] reported detailed observations on the motion of fibres in shear flow of polyacrylamide in corn syrup-water (PAAm) and polyisobutylene in polybutene (PIB-PB) solutions. Their findings were in qualitative agreement with the high Deborah number Oldroyd-B theory of Harlen and Koch, that is, the deviation from Jeffery's orbits can be found in the Boger fluid, i.e., independent of the second normal stress difference. They also found that the period of fibre in viscoelastic fluid is longer than that in Newtonian fluid. However, their observed periods were larger than those predicted from Eq. (57): for PAAm solutions the predicted periods were only 1~3% longer than those for Newtonian fluid but observed to be up to 10% longer; and for PIB-PB solutions, the predicted ones were only 6~25% longer but observed to be 55~84% longer, also see Fig. 6 below. Another observed deviation from the theory is the final orientation of the fibre, which is about 15° away from instead of aligning with the vorticity axis in PAAm solutions, and about 10°~50° in PIB-PB solutions. As shear rate or aspect ratio increases, the long-time behaviour of fibres was significantly deviated from the prediction, even no initial spiralling motion of fibres was observed.

A suitable numerical simulation is an ideal way to provide detailed information on fibres motion in viscoelastic flow. We use the present numerical method to simulate a prolate spheroid rotating in shear flow of the Oldroyd-B fluid. The prolate spheroid is neutrally buoyant in the shear flow with unity shear rate. The aspect ratio of the prolate is 2 with the length of its major axis set at 1; and 384 surface elements with 1536 moving points are used. 25785 fixed field points of are generated in same way to the second field point configuration mentioned previously. This field point configuration was used in simulating the sphere sedimentation and resulted in good results. Though this cannot guarantee its performance in simulating a prolate in shear flow, it is not practical to use more field points with computing facilities currently available to us. The relaxation time of the fluid is set to be 0.70 and the shear rate was 1.0, i.e., Weissenberg number was 0.7. The initial configuration of the configuration tensor was $C(X, 0) = I$ and time step was $0.01.$

The initial orientation vector of the prolate was $(0.50, 0, 0.866)$; i.e., the major axis of the prolate is in the shear and vortex plane (xz) and is inclined at an angle of 60° to the shear direction (x). The relative viscosity is chosen to be from 1.001 to 1.30, i.e., β/β_c from $1/700$ to $3/7$. The simulations were conducted in a cluster of Compaq Alpha workstation and 4 machines were used for each run. The typical CPU time to simulate one time step is about 33 minutes. The Newtonian period, T_0, is 5π when $a_r = 2$ and $\dot{\gamma} = 1.0$. It requires 1571 time steps, i.e. about 864 CPU hours to simulate one period of the prolate's motion when the time increment is $0.01.$ A saving on the computation time was done by simulating only $1/4$ of the period for most of jobs to determine T/T_0. The results would contain some errors due to not fully-established viscoelastic stress. However, the errors may be not too serious for a qualitative comparison with the theory. We checked one typical case of $\eta_r = 1.3$ and $\beta/\beta_c = 3/7$ and found that it took 488 time steps to simulate the first quarter of the period and 1933 time steps to simulate whole period, i.e., $T = 19.33$. If we calculate the period based on the time spent in simulating the first quarter, a figure of $T = 19.52$ is obtained, with an error of less than $1\%.$ The simulated relative delay of particle's rotating due to viscoelastic stress is shown in Fig. 6, with the theory of Harlen and Koch [23] (solid line) and the experimental observation of Iso et al [24] (solid circles and triangles). From the figure, we can see that our numerical results qualitatively agree with the experimental observations and theoretical

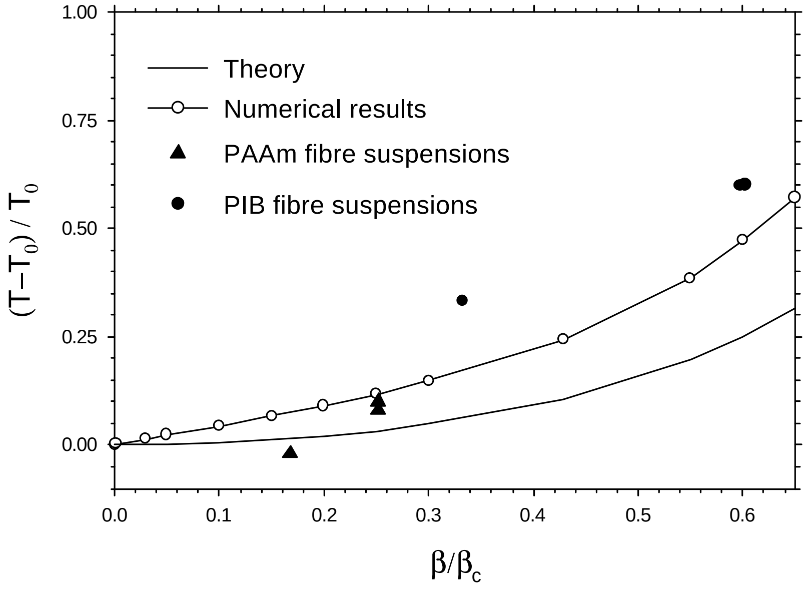

Figure 6: Effect of elasticity on the period of a prolate sheroid in shear flow. The solid line represents the theory of Harlen et al (1993), the symbols the observed results of Iso et al (1996), and open circles (with solid line) our numerical results.

predictions. The viscoelastic stress slowed down the rotation of particles in shear flow. The deviation of the numerical results from the theoretical predictions may be attributed to the basic assumption in the theory, i.e., the Deborah number is much larger than $(\ln(a_r))^{i1}$ and $(\ln(a_r))^{i2} << 1:0$ [23]. In the simulation, Deborah number is less than $(\ln(a_r))^{i1}$ and $(\ln(a))^{i2} > 1:0$: In addition, the theory and experiments deal with fibre suspensions, not a single fibre in isolation. The numerical accumulated errors in a multi-step calculation may be a factor in the accuracy of the numerical solution. As demonstrated in the previous section, insufficient number of field pointsis a key factor contributing to the numerical errors. However, a larger number of field points requires considerable computation efforts, and is not attempted with current computation resources available to us. More accurate comparison would be a subject for further research.

However the present simulation do provide some information on the particles motion in viscoelastic shear flow. The next few figures, we show how the particle rotates and leaves Jeffery's orbit due to viscoelastic stress. The orbits for ´$_r$ = 1:001 and 1:3 (solid lines) are shown in Fig. 7 with Jeffery's orbit (dotted line). When ´$_r$ = 1:001, the viscoelasticity effect is very weak and the fluid is nearly Newtonian. We can see its orbit almost coincides with Jeffery's. When ´$_r$ increases, the orbit deviates from Jeffery's under the driving action of viscoelastic torque. As the viscoelastic torque develops, the particle is driven from Jeffery's to another unclosed orbit. Fig. 8 shows the simulated orbit for Wi = 0:7 and ´$_r$ = 1:3 with the Jeffery's (dotted line). Due to the limitation in computational resources, we only simulate a total time of 30:25, i.e., 3025 time steps. The prolate spheroid is clearly moving along an unclosed orbit. In Fig. 9 the view along the vorticity axis is displayed. The above two figures show that the prolate spheroid is gradually deviating from Jeffery's orbit while the viscoelastic stress is build up and the viscoelastic torque drives the spheroid rotating toward the vortex axis, the z-axis in Figs. 8 and 9.

5 Final Remarks

In this paper, we report the formulation and the implementation of an indirect boundary integral equation method, suitable for solving the mobility problem of a particle system in a viscoelastic fluid. We also present some results of a sphere settling in the Oldroyd-B fluid as a test case and a prolate spheroid in shear flow of an Oldroyd-B fluid. The simulated results are compared with numerical results of Tiefenbruck and Leal [20], the theory of Harlen and Koch [23], and experimental observations of Iso et al [24]. It is demonstrated that the present method is suitable at least for simulating the motion of one particle viscoelastic flow, with currently available computing resources. It is straight forward to use this method to multi-particle systems

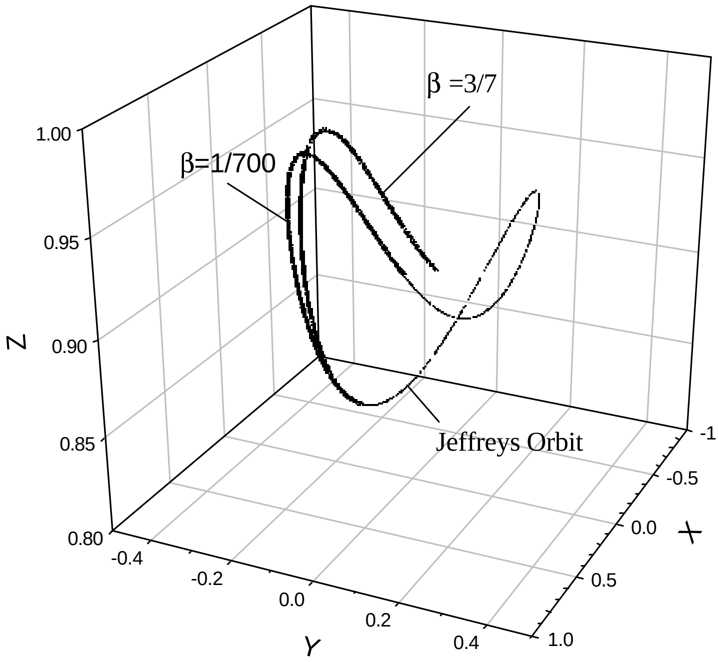

Figure 7: Jeffery's orbit (dotted line) compared to the numerical orbits.

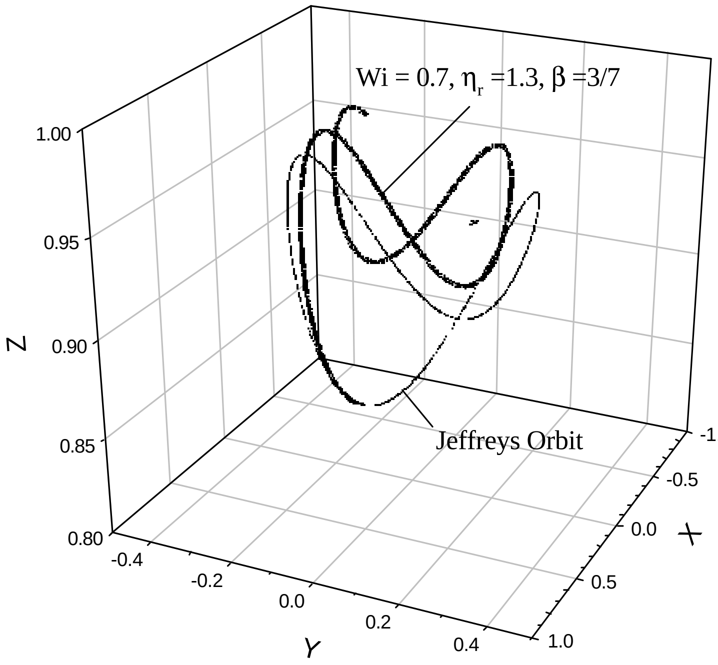

Figure 8: Jeffery's orbit (dotted line) compared to the numerical orbits.

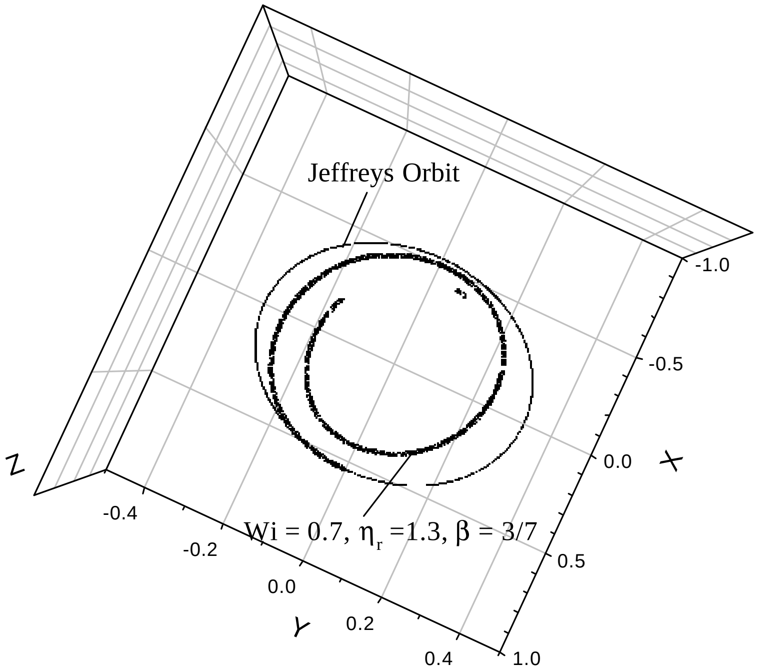

Figure 9: Jeffery's orbit (dotted line) compared to the numerical orbits, as viewed along the vorticity axis.

if more powerful computing resources becoming available.

The features of the present method can be outlined as follows:

- The boundary elements are used to represent the surfaces of the particles, and the boundary conditions on the surfaces can be satisfied more accurately. The boundary element mesh can be easily updated with the particles motion. Computation efficiency is gained from the reduction in dimensionality.
- The radial basis function and the particular solution method are used to avoid volume integration in the boundary integral equation formulation. Hence the volume mesh is not needed in the numerical discretization of the boundary integral equations.
- The complete double layer density formulation of the boundary integral equation with completing and deflation schemes is well-posed for the mobility problems of multi-particle systems. This guarantees the stability of the solution procedure.
- The fixed least square methods are employed in numerical fitting and differentiation without the need of volume meshing.
- A point-wise solver is further developed to solve the constitutive equation for viscoelastic fluid at discrete points in the flow field. All field points move with the mass centre of the particle. Hence, the number of field points can be reduced but the accuracy is maintained. This method requires much less computation and memory than other solvers with mesh or meshless.
- A master/slave programming paradigm using Parallel Virtual Machines (PVM) library software is employed to raise the computation efficiency further.

The present method can be ideally used to flow problems in which inertial and viscoelastic forces do not dominate the flow, i.e., the flow has low Reynolds number and weak elasticity. This is inherent in the iteration process employed in the boundary element formulation.

The boundary integral equation is formulated for the infinite domain but we can only distribute field points within a finite domain in the simulation. The problem is how large the finite domain ought to be and how many field points should be distributed in order to obtain accurate numerical solutions for the infinite domain problem. Tiefenbruck and Leal (1982) used a domain of 10 sphere radii in their axi-symmetric steady simulation of a viscoelastic flow pass a sphere. We are dealing with a 3D time-dependent viscoelastic flow problem with moving boundaries. In addition, the particle has to be tracked for a long time to obtain its trajectory. But we only distribute field points within a cubic box of 10 times the particle size. The replacement of infinite domain by a finite domain results in numerical errors, especially at large Weissenberg number. However, the simulated results agree qualitatively with some theoretical, numerical and experimental results. Hence, this method would find more complicated applications as computer power increases.

Though the present method may be more efficient than some other numerical methods, the computation requirement is still too much for typical present-day computing resources. How to reduce the computation requirements is the key problem of this method. Roughly speaking, the period is increase linearly with the aspect ratio but time increment should decrease with the aspect ratio. The number of field points and boundary elements should increase with the aspect ratio. Even for an aspect ratio of 2, the simulation requires 1600 to 2200 time steps with the time increment of 0:01 to cover one orbital period. A typical CPU time per time step is about 33 minutes for each slave and 19 minutes for the master to simulate a prolate rotating in shear flow with 384 surface element and 25785 field points when 4 machines are used in the Compaq-workstation cluster. The simulations were also conducted in a 3 CPU batch queue of Compaq GS320. The total CPU time per time step is about 68:4 minutes. Hence we could not simulate more complicated problems. But it may be possible to speed up the calculation further. We found that the most CPU time was spent in solving Eq.(31) for the radial basis function using GMRES method. If a suitable preconditioner and more efficient parallel iterative solver can be used, the CPU time can be further reduced.

Acknowledgement The authors appreciate Dr. R. Zheng, Moldflow Corp, for his helpful advices on the particular solution method.

References

[1] S. Kim and S.J. Karrila, Microhydrodynamics: Principles and Selected Applications, Butterworth-Heinemann, Boston, 1991.

[2] A.J. Goldman, R.G. Cox and H. Brenner, Slow viscous motion of a sphere parallel to a plane wall. I. Motion through quiescent fluid, Chem. Eng. Sci., 22 (1967) 637–650.

[3] D.D. Joseph, Y.J. Liu, M. Poletto and J. Feng, Aggregation and dispersion of spheres falling in viscoelastic liquids, J. non-Newt. Fluid Mech., 54 (1994) 45–86.

[4] D.L.E. Becker, G.H. McKinley and H.A. Stone, Sedimentation of a sphere near a wall: weak non-Newtonian and inertial effects, J. non-Newt. Fluid Mech., 63 (1996) 45–86.

[5] P. Singh and D.D. Joseph, Sedimentation of a sphere near a vertical wall in an Oldroyd-B fluid, J. non-Newt. Fluid Mech., 94 (2000) 179–203.

[6] D.D. Joseph and J. Feng, A note on the forces that moves particles in second-order fluid, J. non-Newt. Fluid, 64 (1996) 299–302.

[7] J. Feng, P.Y. Huang and D.D. Joseph, Dynamic simulation of sedimentation of solid particles in an Oldroyd-B fluid, J. non-Newt. Fluid Mech., 63 (1996) 63–88.

[8] D.D. Joseph and Y.J. Liu, Orientation of long bodies falling in a viscoelastic liquid, J. Rheol., 37 (1993) 961–983.

[9] P.Y. Huang, H.H. Hu and D.D. Joseph, Direct simulation of the sedimentation of elliptic particles in Oldroyd-B fluids, J. Fluid Mech., 362 (1998) 297–325.

[10] G.B. Jeffery, The motion of ellipsoidal particles immersed in viscous Fluid, Proc. Roy Soc. Lond, A102 (1922) 161–179.

[11] E.J. Hinch and L.G. Leal, Constitutive equations in suspension mechanics. Part 2. Approximate forms for a suspension of rigid particles affected by Brownian rotations, J. Fluid Mech., 76 (1976) 187–208.

[12] S.H. Dinh and R.C. Armstrong, A rheological equation of state for semi-concentrated fiber suspensions, J. Rheol., 28 (1984) 207–227.

[13] F.P. Folgar and C.L. Tucker, Orientation behavior of fibers in concentrated suspensions, J. Reinforced Plastics and Composites, 3 (1984) 98–119.

[14] N. Phan-Thien and A.L. Graham, A new constitutive model for fibre suspensions: flow past a sphere, Rheol. Acta, 30 (1991) 44–57.

[15] R. Glowinski, T.W. Pan, T.I. Hesla and D.D. Joseph, A distributed Lagrange multiplier/fictitious domain method for particulate flows, Int. J. Multiphase Flows, 25 (1998) 201–233.

[16] N. Phan-Thien and S. Kim, Microstructures in Elastic Media: Principles and Computational Methods, Oxford University Press, New York (1994).

[17] C. Coleman, D. Tullock and N. Phan-Thien, An effective boundary element method for inhomogeneous partial differential equations, ZAMP, 42:(5) (1991) 730-745.

[18] T. Belytschko, Y.Y. Lu and L. Gu, Element-free Galerkin methods, International J. Num. Methods Engig, 37 (1994) 229-256.

[19] C.A. Duarte and J.T. Oden, An $h-p$ adaptive method using clouds, Comput. Methods Appl. Mech. Engrg., 139 (1996) 237-262.

[20] G. Tiefenbruck and L. G. Leal, A numerical study of the motion of a viscoelastic fluid past rigid spheres and spherical bubbles, J. Non-Newt. Fluid Mech., 10 (1982) 115-155.

[21] L. G. Leal, The slow motion of slend rod-like particles in a second-order fluid, J. Fluid Mech., 69 (1975) 305-337.

[22] E. Bartram, H. L. Goldsmith and S. G. Mason, Particle motions in non-Newtonian media III. Further observations in elasticviscous fluids. Rheol. Acta 14 (1970) 776-

[23] O. G. Harlen and D. L. Koch, Simple shear of a suspension of fibres in a dilute polymer at high Deborah number, J. Fluid Mech., 252 (1993) 187-207.

[24] Y. Iso, D. L. Koch and C. Cohen, Orientation in simple shear flow of semi-dilute fibre suspensions 1. Weakly elastic fluids, J. Non-Newtonian Fluid Mech., 62 (1996) 115-134.

[25] T. Tran-Cong and N. Phan-Thien, Three-dimensional study of extrusion processes by boundary element method. Part II. Some non-Newtonian results, Rheologica Acta, 27 (1988) 639-650.

[26] R. Zheng, N. Phan-Thien and C.J. Coleman, A boundary element approach for non-linear boundary-value problems, Computational Mechanics, 8 (1991) 71-86.

[27] T. Nguyen-Thien, Some BIE-based numerical methods for continuum mechanicacs problems, PhD thesis, University of Southern Queensland, March 1999.

[28] E. Oñate, S. Idelsohn, O.C. Zienkiewicz, R.I. Talor and C. Sacco, A stabilized finite point method for analysis of fluid mechanics, Comput. Methods Appl. Mech. Engrg., 139 (1996) 315-346.

[29] J.W. Perram, Statistical Mechanics of hard ellipsoids, I. Overlap algorithm and contact function, J. Comp. Physics, 58 (1985) 409-416.

[30] P. Lancaster and K. Salkauskas, Surface generated by moving least square method. Mathematics of Computation, 37 (1981) 141-158.

[31] J.W. Swegle, S.W. Attaway, M.W. Heinstein, F.J. Mello and D.L. Hicks, An analysis of the smoothed particle hydrodynamics, Technical Report SAND93-2513 UC-705, Sandia National Laboratories, New Mexico, 1994.