Meshless Stochastic Simulation of Micro-Macro Kinetic Theory Models
Elías Cueto, Manuel Laso, Francisco Chinesta

To cite this version:

HAL Id: hal-01007024
https://hal.archives-ouvertes.fr/hal-01007024
Submitted on 2 Apr 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Public Domain
Meshless stochastic simulation of micro-macro kinetic theory models

E. Cueto\textsuperscript{1,*}, M. Laso\textsuperscript{2}, F. Chinesta\textsuperscript{3}

\textsuperscript{1}Group of Structural Mechanics and Materials Modelling (GEMM). Aragón Institute of Engineering Research (I3A).
Betancourt Building. María de Luna, 5.
E-50018 Zaragoza, Spain.

\textsuperscript{2}Laboratory of Non-Metallic Materials
Dept. of Chemical Engineering, Universidad Politécnica de Madrid
Jose Gutierrez Abascal, 2
E-28006 Madrid, Spain

\textsuperscript{3}EADS Corporate International Chair. École Centrale de Nantes
1 rue de la Noé, 44321 Nantes cedex 3, France

Abstract

We present in this paper a numerical technique for the stochastic simulation of molecular models of visco-elastic fluids based on kinetic theory. The technique is based on the use of meshless methods and allows for an updated Lagrangian description of the conservation equations. It makes use of Natural Neighbor Galerkin schemes, that allow for a proper geometrical description of the domain as it evolves. The presented technique is especially well suited for the numerical simulation of free-surface flows. In this way, model molecules are associated with nodal positions such that they are advected with material velocities. Problems associated with lack of molecules in certain elements, for instance, as encountered in the basic implementation of CONNFFESSIT approaches, are thus avoided.

We present examples of validation and also performance tests of this technique applied to FENE and reptation (Doi-Edwards) models.

\textsuperscript{*}Correspondence to: Elías Cueto. Mechanical Engineering Department. Edificio Betancourt. University of Zaragoza. María de Luna, 5. E-50018 Zaragoza, Spain. e-mail: ecueto@unizar.es
1 Introduction

Models arising from kinetic theory are nowadays a useful technique for the derivation of complex constitutive equations for the numerical simulation of visco-elastic flows. In fact, the majority of constitutive equations used in continuum numerical simulations of such fluids can be derived from kinetic theory [28]. However, the derivation of a constitutive equation from kinetic theory models often involves closure approximations, that can have significant impact on the final predictions of the model thus constructed [11][27][33].

Micro-macro techniques aimed at resolving this problem. Although much more computationally expensive, these techniques, that couple coarse-grained molecular scale of kinetic theory to the macro scale of the continuum, have been used increasingly in the last years [29] [20]. In it, advantage is obtained by exploiting the equivalence of the Fokker-Planck equation, that describes the evolution of the probability distribution function governing the configuration state at the micro level, and an Itô stochastic differential equation. Instead of solving the deterministic Fokker-Planck equation, an equivalent stochastic differential equation is solved by means of a large enough ensemble of realizations of the stochastic process. The viscoelastic stress is thus obtained as an ensemble average of some function of the micro state at each molecule.

In the original CONNFFESSIT approach, the simulation begins by solving conservation equations, generally by using standard Eulerian or ALE Finite Element Methods. With the updated velocity field, a set of molecules are distributed on the flow, and their trajectory is computed in a Lagrangian framework. The Itô’s stochastic differential equation is then integrated along each particle’s trajectory and finally the viscoelastic stress is obtained by averaging the configuration state of the particles found at each finite element.

This approach presents, despite its success for many applications, some drawbacks (see the review by Keunings [28] and references therein). First, molecules must be located within the finite element mesh. This tracking may be expensive for complex geometries. Second, a large number of particles must be located within each finite element in order to provide an accurate enough visco-elastic stress prediction, free of statistical noise.

These problems can be avoided if a Lagrangian strategy is implemented. In it, molecules are attached to the nodes, travelling together with material velocity. In this way, no element runs out of particles and no tracking algorithm is necessary to determine the element a particle is located in. The obvious problem associated with Lagrangian strategies is mesh distortion, or remeshing, if performed, and its associated numerical diffusion. However, in the middle nineties, the family of meshless methods [8] gained a tremendous
.popularity due precisely to their lack of sensitivity to mesh distortion. No mesh (in a FE sense) must be generated in meshless methods, since it is done by the method in a process transparent to the user.

Meshless methods thus opened the possibility of Lagrangian approaches in many branches of Computational Mechanics [32] [31] [1] [22]. In this paper we analyze how these methods can help to solve some of the problems associated to stochastic approaches to micro-macro models of visco-elastic fluid flow.

To this end, we have employed the Natural Element Method (NEM) [38]. This method offers some advantages over other meshless methods for this kind of applications. First of all, the enforcement of essential boundary conditions is straightforward if we employ the α-NEM [15] or C-NEM [41] versions of the method. Second, tracking of free surface is also straightforward. α-NEM techniques notably simplify this issues such that no boundary marker or similar is necessary. Thus, the use of techniques such as Volume of Fluid (VoF) or similar, employed recently in the context of visco-elastic free surface flows [9] can also be overcome.

The outline of the paper is as follows. In section 2 we describe the basics of the NEM, together with its most important properties. In section 3 we describe the models implemented, namely FENE and reptation models, and finally in section 4 we present some numerical results that demonstrate the potential of the method. The paper is completed with the conclusions.

2 The Natural Element Method

As mentioned before, one of the main advantages of meshless methods in general, and NEM in particular, is that no lack of accuracy is due to mesh distortion [38]. This is in sharp contrast with the Finite Element Method, whose behavior under distortion is well known many years ago [7].

In essence, the NEM is a Galerkin method, just like the FEM, but instead of piece-wise polynomial approximation, natural neighbor (and hence its name) approximation is used. In this section the construction of natural neighbor interpolation is reviewed, together with its most important properties. Prior to this definition we introduce some basic geometrical entities that are needed for further developments.

2.1 Voronoi/Dirichlet diagrams

Consider a model composed by a cloud of points \( N = \{n_1, n_2, \ldots, n_m\} \subset \mathbb{R}^d \), for which there is a unique decomposition of the space into regions such that
each point within these regions is closer to the node to which the region is associated than to any other in the cloud. This kind of space decomposition is called a Voronoi diagram (also Dirichlet tessellation) of the cloud of points and each Voronoi cell is formally defined as (see figure 1):

$$T_I = \{ x \in \mathbb{R}^d : d(x, x_I) < d(x, x_J) \forall J \neq I \},$$

(1)

where $d(\cdot, \cdot)$ is the Euclidean distance function.

The dual structure of the Voronoi diagram is the Delaunay triangulation\(^1\), obtained by connecting nodes that share a common $(d-1)$-dimensional facet. While the Voronoi structure is unique, the Delaunay triangulation is not, there being some so-called degenerate cases in which there are two or more possible Delaunay triangulations (consider, for example, the case of triangulating a square in 2D, as depicted in Fig. 1 (right)). Another way to define the Delaunay triangulation of a set of nodes is by invoking the empty circumcircle property, which means that no node of the cloud lies within the circle covering a Delaunay triangle. Two nodes sharing a facet of their Voronoi cell are called natural neighbours and hence the name of the technique.

Equivalently, the second-order Voronoi diagram of the cloud is defined as

$$T_{IJ} = \{ x \in \mathbb{R}^d : d(x, x_I) < d(x, x_J) < d(x, x_K) \forall J \neq I \neq K \}. \quad (2)$$

Based on these definitions, different natural neighbour interpolation schemes have been proposed. We focus on Thiessen and Sibson interpolation [36]. The interested reader can look for other related interpolation schemes in [39] and references therein.

\(^1\)Even in three-dimensional spaces, it is common to refer to the Delaunay tetrahedralisation with the word triangulation in the vast majority of the literature.
2.2 Thiessen interpolation

The simplest of the natural neighbour-based interpolants is the so-called Thiessen’s interpolant [40]. Its interpolating functions are defined as

\[ \psi_I(x) = \begin{cases} 
1 & \text{if } x \in T_I \\
0 & \text{elsewhere} 
\end{cases} \tag{3} \]

The Thiessen interpolant is a piece-wise constant function, defined over each Voronoi cell. It defines a method of interpolation often referred to as nearest neighbour interpolation, since a point is given a value defined by its nearest neighbour. Although it is obviously not valid for the solution of second-order partial differential equations, it can be used to interpolate the pressure in formulations arising from Hellinger-Reissner-like mixed variational principles, as proven in [23].

2.3 Sibson’s interpolation

The most extended natural neighbour interpolation method, however, is the Sibson interpolant [36] [37]. Consider the introduction of the point \( x \) in the cloud of nodes. Due to this introduction, the Voronoi diagram will be altered, affecting the Voronoi cells of the natural neighbours of \( x \). Sibson [36] defined the natural neighbour coordinates of a point \( x \) with respect to one of its neighbours \( I \) as the ratio of the cell \( T_I \) that is transferred to \( T_x \) when adding \( x \) to the initial cloud of points to the total volume of \( T_x \). In other words, if \( \kappa(x) \) and \( \kappa_I(x) \) are the Lebesgue measures of \( T_x \) and \( T_{xI} \) respectively, the natural neighbour coordinates of \( x \) with respect to the node \( I \) is defined as

\[ \phi_I(x) = \frac{\kappa_I(x)}{\kappa(x)}. \tag{4} \]

In Fig. 2 the shape function associated to node 1 at point \( x \) may be expressed as

\[ \phi_1(x) = \frac{A_{abcdef}}{A_{abcdef}}. \tag{5} \]

2.4 Properties of Sibson’s interpolation

Sibson interpolants have some remarkable properties that help to construct the trial and test functional spaces of the Galerkin method (see [38] for proofs of the following properties).
Besides properties like continuity and smoothness (everywhere except at the nodes), Sibson interpolants possesses linear completeness (i.e., exact reproduction of a linear field).

Sibson interpolants can also reproduce linear functions exactly along convex boundaries. This is in sharp contrast to the vast majority of meshless methods. In addition, in [15] [14] [41] distinct methods of imposing linear displacement fields along non-convex boundaries were developed. These are based on the use of $\alpha$-shapes, $\varepsilon$-samplings or visibility criteria, respectively. So, essential boundary conditions can be imposed directly, as in traditional Finite Element methods.

In [15] and [13] it was demonstrated that the construction of the Sibson interpolant over an $\alpha$-shape [17] of the domain allows us to accurately extract the shape of the domain, defined in terms of nodes only, while ensuring linear interpolation along any kind of boundaries (convex or not). This property was later generalized for arbitrary clouds of points and a explicit definition of the domain through CAD techniques in [14].

In the next section we study the implication of $\alpha$-shapes in the development of the method here proposed.

3 The $\alpha$-shapes-based Natural Element Method

The identification of the free surface in an updated Lagrangian flow simulation deserves some comments. In many prior works, location of boundary nodes is performed by flagging coincident element faces [30], for instance. Once the updating of nodal positions has been performed, a recursive check must be done in order to find overlapping boundary segments. In three di-
mensions this technique becomes prohibitive, since the markers should be boundary triangles.

In meshless methods models are constructed by a set of nodes only, and thus boundary tracking is usually performed by employing different strategies, in absence of such information. In particular, we have employed *shape constructors* to perform this task [22]. Shape constructors are geometrical entities that transform finite point sets into a multiply connected shape in general. Due to their importance in many areas, they have attracted much attention in Computational Geometry in the last years. In particular, we employ $\alpha$-shapes [17]. $\alpha$-shapes define a one-parameter family of shapes $S_\alpha$ (being $\alpha$ the parameter), ranging from the “coarsest” to the “finest” level of detail. $\alpha$ can be seen, precisely, as a measure of this level of detail. This means that all details of size less than $\alpha$ will be ignored in the geometry of the domain. This is a common practice in many conversions from CAD models (very often geometrically “exact”) to FEM models, in which very small features of the geometry are ignored from a mechanical point of view.

An $\alpha$-shape is a polytope that is not necessarily convex nor connected, being triangulated by a subset of the Delaunay triangulation of the points. The Delaunay triangulation is chosen as a departure point due to its uniqueness (other than degenerate cases). But the Delaunay triangulation is always defined on the convex hull of the cloud of nodes, and thus no associated shape is obtained. A filtering is then applied to the list of triangles (tetrahedra) on the triangulation. They are ordered according to the radius of their circumcircle (circumsphere).

If a level of detail $\alpha$ is then fixed, it is clear that, if the nodal sampling is dense enough —for this, taking $h$, the nodal spacing parameter, lower than $\alpha$ will suffice—, all triangles whose circumcircle is bigger than $\alpha$ must be eliminated from the true geometry of the domain. Crudely speaking, they hide concave features of the geometry, by assumption bigger than $\alpha$.

The theory of $\alpha$-shapes was soundly established in [16] for points in the plane and later in [17] for three-dimensional clouds. $\alpha$-shapes constitute a provable geometry reconstruction algorithm, in the sense that there exist formal geometrical and topological guarantees on the final result, based on the size of the sampling, as mentioned above. The interested reader should consult [17] for more details.

In Fig. 3 an example of the just introduced theory is presented. It represents some instances of the finite set of shapes for a cloud in an intermediate step of the simulation of a wave breaking in a beach. Note how as $\alpha$ increases from Fig. 3(a) to (f), we obtain different shapes ranging from the cloud of nodes itself (for $\alpha = 0$) to the convex hull of the cloud. For initial value problems, in which we know the starting configuration of the cloud, and
Figure 3: Evolution of the family of $\alpha$-shapes of a cloud of points representing a wave breaking on a beach. Shapes $S_0$ (a), $S_{0.5}$ (b), $S_{1.0}$ (c), $S_{2.0}$ (d), $S_{3.0}$ (e) and $S_\infty$ (f) are depicted.
thus the exact geometry of the domain at the beginning of the simulation, but not the subsequent geometries being described for different time steps, we employ a related strategy. We first fix the nodal sampling (so to speak, the nodal spacing parameter \( h \), according to the needs of the simulation) and then fix \( \alpha > h \). All details of size slightly bigger, but close to, \( h \) will then be reproduced, as in Fig. 3(d). More details on this technique can be found in [19] and [22]. Obviously, in large geometrical transformations an \( \alpha \)-adaptivity can be envisaged [15].

4 Mechanical modelling

Following Keunings [28], kinetic theory possesses three main ingredients. One is the configuration space, denoted here by \( X \). It represents the set of all possible values for molecular configurations. The second ingredient characterizes the state of all the molecules at a given point in the model, through a probability distribution function, \( \psi(X, x, t) \). Under equilibrium conditions we assume that this distribution function takes the form \( \psi_{eq} \). In these conditions, Brownian and elastic forces are, of course, in equilibrium.

Non-equilibrium situations provoke in the fluid the appearance of a visco-elastic stress, denoted by \( \tau \). Velocity and stress fields are linked by the following conservation equations, in absence of inertia terms:

\[
\text{Div} \sigma = 0 \quad (6) \\
\text{Div} v = 0 \quad (7) \\
\sigma = -pI + \tau + 2\eta \dot{\gamma} \quad (8)
\]

where \( \dot{\gamma} \) represents the strain rate tensor, \( \dot{\gamma} = \frac{1}{2}(\nabla v + \nabla v^T) \), \( \eta \) the solvent viscosity (when it exists) and \( \tau \) the polymer stress coming from the microscopic model. The resulting model defines a two-scales non-linear transient model because the non-linear and transient behavior of the microscopic description that we introduce later.

The simplest solution algorithm lies in the use of an explicit strategy that allows linearizing the microscopic model by decoupling both description scales. Thus, the weak formulation related to Eqs. (6) and (8) when the extra-stress \( \tau \) is assumed known from the previous time step \( (\tau^{n-1}) \) writes:

\[
\int_{\Omega} 2\eta \dot{\gamma}^n : \dot{\gamma}^* d\Omega - \int_{\Omega} p^n I : \dot{\gamma}^* d\Omega + \int_{\Omega} \tau^{n-1} : \dot{\gamma}^* d\Omega = 0 \quad (9)
\]

In absence of a solvent contribution, i.e. when \( \eta = 0 \) in Eq. (9), as encountered in usual models of entangled polymers and melts, a DEVS formulation [5] is retained.
To complete the formulation, kinetic theory provides also the equation governing the evolution of the probability distribution function \( \psi \). This equation is known as the Fokker-Planck equation:

\[
\frac{D\psi}{Dt} = -\frac{\partial}{\partial X} \cdot \{A \psi\} + \frac{1}{2} \frac{\partial}{\partial X} \frac{\partial}{\partial X} : \{D \psi\} \tag{10}
\]

where \( A \) represents a vector describing the drift exerted by the fluid on the function \( \psi \), and \( D \) is a symmetric, positive-definite matrix accounting for brownian effects in the model. \( D/Dt \) represent material derivative. The expression, finally, that relates the obtained configuration state with its enforced state of stress is known as the Kramers formula:

\[
\tau = \int g(X) \psi dX = \langle g(X) \rangle \tag{11}
\]

where the brackets \( \langle \cdot \rangle \) denote an ensemble average over all the molecular conformational space at a physical point, and \( g \) is some function of the configuration state, depending on the particular model considered. Despite the existence of some incipient deterministic schemes able to solve efficiently Eq. (10) (see [4] [2] [3] [35] [12] [18] and the references therein), stochastic simulation strategies are nowadays much more used, fact that justifies the choice retained in this work.

The stochastic approach to the problems takes benefit from the equivalence of the Fokker-Planck equation (10) to the following Itô’s stochastic differential equation [25]:

\[
dX = A dt + B \cdot dW \tag{12}
\]

where

\[
D = B \cdot B^T \tag{13}
\]

and \( W \) represents a Wiener process. Equation (12) applies along individual molecule trajectories. Thus, in our approach, these molecules will be attached to the nodes, allowing for the integration of Eq. (12) by the method of characteristics, along the nodal paths. The integration is performed by means of the so-called Euler-Maruyama scheme:

\[
X_{n+1}^j = X_n^j + A(X_n^j, t_n) \Delta t + B(X_n^j, t_n) \cdot \Delta W_n^j \tag{14}
\]

where \( n \) refers to the current time step and \( j \) to the individual molecule being integrated.

In this paper two different models based on the stochastic approach to the Fokker-Planck equations have been implemented. The first one is the well-known FENE model for polymer solutions and the second one is the Doi-Edwards model of entangled polymers based on the repatation concept.
4.1 FENE model

The Finite Extension Non-linear Elastic (FENE) model of polymers represents a set of elastic dumbbells that do not interact with each other, flowing in a Newtonian solvent. Each dumbbell is composed by two Brownian beads joined by a spring. Thus, the conformation space is given by $X = Q$, $Q$ being the vector joining the beads. This spring is assumed to have a non-linear behavior, such that it has a finite, limited extension $Q_0$. Thus, the constitutive equation for these springs will be [34]:

$$F_c = \frac{H}{1 - Q^2/Q_0^2}Q$$

with $H$ the spring constant. Each bead is subjected to an hydrodynamic drag given by

$$F^h = -\zeta [(\dot{r}) - (\kappa \cdot r + v_0)]$$

where $r$ represents the position vector of each bead, $\kappa$ the gradient of the velocity field and then $\kappa \cdot r + v_0$ denotes the undisturbed fluid velocity at the bead position (homogeneous deformations are assumed at the molecule scale). Finally, $\zeta$ represents the viscous friction coefficient existing between the beads and the solvent. Moreover, the beads are subjected to Brownian forces originated by the bombardment of solvent molecules

$$F^b = -kT \left( \frac{\partial \ln \Psi}{\partial \dot{Q}} \right)$$

The dimensionless stress tensor for the FENE model is given by

$$\tau = -nk_BT \left( \frac{\dot{Q} \dot{Q}}{1 - \hat{Q}^2/b} \right) + I$$

where, in turn, $\hat{Q} = Q/(k_BT/H)^{1/2}$, $k_B$ represents the Boltzmann constant. $b$ represents the so-called “dimensionless finite extensibility parameter” given by $b = HQ_0^2/k_BT$.

The dimensionless Fokker-Planck equation for this particular model reads [26]:

$$\frac{\partial \hat{\psi}}{\partial \hat{t}} = -\frac{\partial}{\partial \hat{Q}} \cdot \left( \frac{\hat{\kappa} \cdot \hat{Q} \hat{\psi} - \frac{1}{2} \frac{1}{1 - \hat{Q}^2/b} \hat{Q} \hat{\psi}}{2} + \frac{1}{2} \frac{\partial}{\partial \hat{Q}} \cdot \frac{\partial}{\partial \hat{Q}} \hat{\psi} \right)$$

In this last equation, $\hat{\kappa}$ represents the dimensionless form of the velocity gradient and $\hat{t} = t/\lambda_H$ with $\lambda_H = \zeta/4H$ a characteristic time of the polymer.
The dimensionless distribution function $\hat{\psi}$ is given by

$$\hat{\psi}(\hat{Q}, \hat{t}) = \frac{\psi(Q, t)}{(k_B T)^{3/2}}$$  \hspace{1cm} (20)$$

The equivalent Itô’s stochastic differential equation for this model, Eq. (12), will be given by

$$d\hat{Q}_i = \left( \hat{\kappa} \cdot \hat{Q}_i - \frac{1}{2} \hat{Q}_i^2 / b \right) d\hat{t} + dW_i$$  \hspace{1cm} (21)$$

In this case, $W_i$ is a three-dimensional Wiener process. To numerically integrate this equation we employed the second-order semi-implicit predictor-corrector algorithm proposed in [25] and also employed in [26]. Further details can be found in these references.

### 4.2 Doi-Edwards model

The reptation model implemented in this work is essentially the ubiquitous Doi-Edwards or Curtiss-Bird model, see [25] and references therein. In it, the configuration space $X$ is defined by a tube segment orientation vector $u$ and the normalized contour label $s \in [0, 1]$. Thus, the probability distribution function $\varphi$ will now be function of $u$, $s$, $x$ and $t$. The expression $\varphi(u, s, x, t) du ds$ gives the probability of finding at time $t$ and position $x$ a tube segment oriented between $u$ and $u+du$ and containing the chain segment $[s, s+ds]$. The Fokker-Planck equation associated with this model reads

$$D\psi \frac{D}{Dt} = -\frac{\partial}{\partial u} \cdot [(I - uu) \cdot \kappa \cdot u\psi] + \frac{1}{\tau_d} \frac{\partial^2 \psi}{\partial s^2}$$  \hspace{1cm} (22)$$

with $\tau_d$ the so-called disengagement or reptation time [28]. The constitutive Kramers equation has been taken in this work as

$$\tau = G\langle uu \rangle$$  \hspace{1cm} (23)$$

The Fokker-Planck equation is now equivalent to two different Itô stochastic differential equations. The first one, which is actually deterministic, reads

$$du = (I - uu) \cdot \kappa \cdot u dt$$  \hspace{1cm} (24)$$

stating that the evolution of the tube orientation is governed by the drift of the surrounding fluid, through its velocity gradient, $\kappa$. The second one,

$$ds = \sqrt{\frac{2}{\pi^2 \tau_d}} dW$$  \hspace{1cm} (25)$$

governs the evolution of the chain segment escaping the tube, and is purely diffusive. Again, $W$ represents a Wiener process.
5 Model integration

5.1 Functional approximation

The variational form of the macroscopic description of the flow kinematics, given by a Hellinger-Reissner-like principle, see Eq. (9), is approximated by choosing suitable interpolation schemes for both velocities and pressure fields. It is well-known that not all approximations give rise to a stable method. The well-known LBB condition \[6\] is recognized as the condition to be fulfilled by such an approximation.

In this work we have employed a mixed Sibson-Thiessen approximation for velocities and pressures, respectively. Although it is known that this kind of approximation does not verify the LBB condition for some, very specific, cases (see [23]), it often renders stable results and is very convenient from the computational point of view. Other approximations are also possible within the NEM. For instance, in [23] an enriched approximation that verifies the LBB condition was presented, although it includes one degree of freedom more for each node. Recently, natural neighbor approximations with quadratic or higher order consistency have been also presented, see [24][42]. Both possibilities give a stable approximation for this kind of variational formulations.

5.2 Model discretization

Once an appropriate interpolation scheme has been chosen, the variational form of the problem given by Eqs. (6)-(8) is approximated in a Bubnov-Galerkin framework,

\[
\begin{align*}
  v^h(x) &= \sum_{l=1}^{n} \phi_l(x)v_l \\
  p^h(x) &= \sum_{l=1}^{n} \psi_l(x)p_l.
\end{align*}
\]

where \( n \) represents the number of nodes considered in the approximation (natural neighbors of point \( x \)). The functions \( \psi_l(x) \) and \( \phi_l(x) \) in this work represent some form of natural neighbour interpolation, as mentioned before (Thiessen and Sibson, respectively, for the choice made in this work). This leads to the following system of algebraic equations:

\[
\begin{pmatrix}
  \bar{K} & G \\
  G^T & M
\end{pmatrix}
\begin{pmatrix}
  v \\
  p
\end{pmatrix} =
\begin{pmatrix}
  f \\
  0
\end{pmatrix}
\]
where

\[ K_{IJ} = \int_{\Omega} B_I^T \mathcal{C} B_J d\Omega \]  (29)

\[ G_{IJ} = -\int_{\Omega} \tilde{B}_I^T \psi_J d\Omega \]  (30)

\[ M_{IJ} = -\frac{1}{\lambda} \int_{\Omega} \psi_I \psi_J d\Omega \]  (31)

\[ f_I = \int_{\Gamma_t} \phi_I T d\Gamma \]  (32)

where \( \mathbf{7} \) represents the applied tractions on the boundary and

\[ \tilde{B}_I = [\phi_{I,1}(x) \phi_{I,2}(x)] \]  (33)

\[ B_I = \begin{pmatrix} \phi_{I,1}(x) & 0 \\ 0 & \phi_{I,2}(x) \end{pmatrix} \]  (34)

\[ \mathcal{C}_{IJKL} = \eta(\delta_{IK}\delta_{JL} + \delta_{IL}\delta_{JK}) \]  (35)

In the Eq. (33), \( \phi_{I,1} \) and \( \phi_{I,2} \) represent the derivatives of the shape function with respect to coordinates 1 and 2, i.e., \( x \) and \( y \), respectively. It must be noted that, if we consider totally incompressible situations, \( M = 0 \).

### 5.3 An overview of the solution algorithm

The proposed procedure implies some modifications to the standard CONNFFESSIT technique. The standard algorithm, as proposed by [29], was composed of the following steps:

- **Step 1**: Solve the macro equations via FEM and obtain the velocity field \( \mathbf{v} \)
- **Step 2**: Using \( \mathbf{v} \), compute the path of each molecule
- **Step 3**: Integrate the S.D.E. governing the micro level
- **Step 4**: Update the viscoelastic stress by means of averaging

To this end, a large enough number of particles are distributed on the geometry of the flow and convected with it. Keunings [28] cites some difficulties associated to this procedure, such as the tracking of particles. Simple schemes for tracking particles can be highly inaccurate. In addition, it can be highly CPU demanding, due to the large number of particles needed to
avoid numerical noise (in the order of $10^5$–$10^7$). Finally, to avoid statistical problems, each finite element must contain a minimum number of particles ($10^2$–$10^3$), and this could not happen, depending on the conditions of the flow.

The proposed algorithm, however, proceeds as follows:

- Step 1: Solve the macro equations via NEM and obtain the velocity field $\mathbf{v}$
- Step 2: Particles’ path is nodal path, so advancing their position is straightforward
- Step 3: For each node, integrate the S.D.E. and update configuration
- Step 4: Update the viscoelastic stress by means of averaging

The advantage of this procedure is two-fold. First, no tracking algorithm is necessary, since particles are attached to the nodes. Second, no element runs out of particles. The price to pay is to obtain the velocity gradient at nodal position, in order to perform Step 3 below. This is addressed in the next section.

### 5.4 Numerical issues

The correct implementation of the models described above requires to take into consideration the best way to compute numerically the velocity gradient, $\kappa$, at nodal positions. This is needed, since we assume that model’s particles will be attached to the nodes and travel with material velocity along the characteristic lines. Three main possibilities arise, in view of the particular features of Sibson’s interpolation, presented in section 2. This kind of interpolation is not differentiable at nodal positions (similarly to finite elements), and thus the velocity gradient is not defined at these positions.

One first possibility is to employ standard Gauss quadratures on Delaunay triangles and to project an average of these values to the nodes. To this end we employ the superconvergent patch recovery approach [43], slightly modified to fit into NE methods. Let $\kappa^*$ be a matrix containing the desired nodal values of the velocity gradient at nodes neighboring a particular Delaunay triangle. We obtain a smoothed gradient field by interpolating these values through

$$\kappa^* = \Phi \kappa^*$$ (36)
where $\Phi$ represents the matrix containing the NEM’s shape functions values. The problem is now formulated as the minimization of the functional

$$
\int_{\Omega} (\kappa^* - \kappa^h)^2 d\Omega
$$

(37)

with $\kappa^h$ the gradient field obtained by direct differentiation of the velocity field, and thus discontinuous. Here, by $\Omega$, we denote the Delaunay triangle. After minimization,

$$
\kappa^* = \frac{\int_{\Omega} \Phi \kappa^h d\Omega}{\int_{\Omega} \Phi^T \Phi d\Omega}
$$

(38)

We employ three-point Gauss quadrature on triangles. Since the number of natural neighbors of any point is usually more than three, the resulting system of equations is often ill-behaved. However, by employing conjugate gradient methods, for instance, in solving Eq. (38) we have found no problem, if the algorithm is feeded with an estimate of the result (the values at Gauss points, for instance, which should be close to the sought nodal values).

Once equation system (38) has been solved, different values of $\kappa^*$ are predicted for each node belonging to neighboring triangles. The process finishes by averaging these values at each node.

This technique is perhaps the crudest one, but it is computationally less expensive than the other two, namely, the employ of stabilized nodal quadratures [10] [21] —which assumes a homogenized gradient value on each Voronoi cell— or the use of natural neighbor interpolants of continuity $C^1$ or higher. In this last case, the derivative of the velocity field would be obtained by simply differentiating the method’s shape functions at the nodal positions. The interested reader can consult [24] for more details in this last method.

We employed the first approach in order to minimize the computational cost of the global method, which is high by nature due to the stochastic character of the equations.

6 Numerical results

6.1 Validation of the proposed technique: start-up of shear flow

In order to validate the method proposed before, we have reproduced well-known results from existing techniques. To this end, a homogeneous shear flow was considered given by $v_x = \dot{\gamma} y$, $v_y = v_z = 0$, imposed at the nodes on the boundary of a squared domain.
The results obtained in [26] by employing a standard CONNFFESSIT approach for the FENE model were compared to ours. A cloud of 525 nodes was employed. A uniform velocity gradient was imposed at the integration points and the resulting stress field was analysed. The projection technique explained in Section 5.4 was employed. The objective is thus to validate the projection algorithm, as well as the numerical integration for the SDE.

Results for the evolution of the viscosity, first and second normal stress coefficients are shown in Fig. 4 for one particular node in the cloud. Results are similar for all nodes. In all cases, $\lambda_H \cdot \dot{\gamma} = 100.0$ and $b = 20.0$.

In general, good agreement is observed with respect to the results report in the detailed study by Herrchen and Ötinger [26]. For long times, a somewhat more pronounced statistical noise is observed, due to fact that a limited number of stochastic realizations has been accomplished at each node. In this case, 10000 realizations were considered. Of course, this is not the best approach to simulate a homogeneous, rheometrical flow, but it was considered here as a means to validate the approach. More complex, non-homogeneous flows, with free surfaces, are considered in the following sections.

### 6.2 Die exit flow of a FENE fluid

In this section we consider the time evolution of a die swelling flow of a FENE fluid. The initial cavity and cloud of points is depicted in Fig. 5.

The flow of the fluid is forced by adding new nodes entering the channel with prescribed parabolic velocity. The maximum value of this inflow velocity was set to $0.1 \text{m/s}$ and $\lambda_H = 1.0$. These nodes incorporate particles with equilibrium probability distribution function $\psi_{eq}$. Thus, the number of nodes in the model is increasing with time. The time step during the simulation, constant, was set to 0.005s, both for the flow kinematics solver and the Euler-Maruyama integration scheme for the SDEs. This was so for simplicity, but different strategies can also be implemented. For instance, updating nodal positions only after some time steps in the integration of the SDEs is also possible, and sometimes even convenient, given the small time increment needed for a proper result in the Euler-Maruyama scheme.

In Fig. 6 a snapshot of the velocity field for the initial and an intermediate time step are shown. In it, the elastic behaviour of the fluid is pointed out. In this case, some minor loss of symmetry is observed. This reveals that the number of particles associated to each node is not large enough. We verified that by increasing the number of molecules the symmetry is enhanced.

There is another reason for this asymmetry of a more geometrical nature. We employed three-points integration on the triangles. As is well-known,
Figure 4: Evolution of non-dimensional viscosity (a), first stress normal coefficient (b) and second normal stress coefficient (c) in a FENE fluid undergoing the startup of shear flow.
Delaunay triangulation for four nodes lying on the same circle is not unique, and the use of a very regular node distribution at the early stages of the simulation can lead to non-symmetric triangulations at both sides of the symmetry axes. This can be minimised by using Stabilised Conforming Nodal Integration techniques [31].
6.3 Die swelling for the Doi-Edwards model

The same geometry used in Section 6.2 is again employed for the simulation of
die swelling of an entangled polymer modelled with the Doi-Edwards theory
presented above. In this case, a time step of 0.007s was employed with
\( \lambda_H = 1.0 \), and only 100 molecules were attached to each node to alleviate
the computational cost involved in this kind of micro-macro simulations.

Four snapshots of the velocity field are shown in Fig. 7. Again, elastic
effects are notorious after the outlet of the channel. A minor loss of symmetry
in the flow is noticed due mainly to the statistical noise. Despite the very
low number of realisations per node, the statistical noise remains surprisingly
low.

7 Conclusions

A method for the simulation of complex models arising from kinetic theory
of fluids has been presented. This method is based on the combination of two
different ingredients. On one hand, the use of a CONNFFESSIT-like strat-
tegy, in which benefit is obtained from the equivalence of the Fokker-Planck
equation and an Itô’s stochastic differential equation. Secondly, an updated
Lagrangian approach by employing the Natural Element Method. This mesh-
less strategy allows to associate the particles carrying the realizations of the
stochastic process to the nodes. They will consequently move with material
velocity, avoiding the need of complex particle tracking algorithms and the
related numerical problems (lack of particles in certain regions of the domain,
etc.) The technique is used in conjunction with shape constructors to track
the evolution of the free surface.

A comparison with existing results for FENE models has been presented.
It shows good accuracy, with respect to results existing in the literature for
homogeneous flows. Some examples of free-surface die swelling flows with
both FENE and Doi-Edwards models have also been presented. Although
no closed-form results exist for these problems, the results obtained show a
priori a promising potential of the proposed technique in the prediction of
complex, non-homogeneous flows in the presence of free surfaces.

The main drawback of the technique is, maybe, the computational cost.
In order to keep the number of degrees of freedom reasonably low, a moderate
number of realisations of the stochastic process should be accomplished. The
price to pay is an obvious increase in the numerical noise of the results, that
can eventually lead to some minor loss of symmetry in certain simulations.
The FENE model has demonstrated to be more sensitive to this effect than
Figure 7: Evolution of the velocity field in the die swelling flow of reptating polymer.
the Doi-Edwards model. We believe, however, that the presented technique constitute a valuable alternative to traditional finite elements, since it avoids both the mesh distortion effects associated to Eulerian and ALE approaches, and the numerical diffusion associated to remeshing. In any case the stochastic integration that constitutes the most expensive computational operation can be carried out in parallel using appropriate computing platforms.

Acknowledgements

This work has been partially supported by the Spanish Ministry of Science and Innovation, through grant number CICYT-DPI2008-00918. This support is gratefully acknowledged.

References


