

Chapter 2

A Lagrangian–Lagrangian Framework for the Simulation of Rigid and Deformable Bodies in Fluid

Arman Pazouki, Radu Serban and Dan Negrut

Abstract We present a Lagrangian–Lagrangian approach for the simulation of fully resolved Fluid Solid/Structure Interaction (FSI) problems. In the proposed approach, the method of Smoothed Particle Hydrodynamics (SPH) is used to simulate the fluid dynamics in a Lagrangian framework. The solid phase is a general multibody dynamics system composed of a collection of interacting rigid and deformable objects. While the motion of arbitrarily shaped rigid objects is approached in a classical 3D rigid body dynamics framework, the Absolute Nodal Coordinate Formulation (ANCF) is used to model the deformable components, thus enabling the investigation of compliant elements that experience large deformations with entangling and self-contact. The dynamics of the two phases, fluid and solid, are coupled with the help of Lagrangian markers, referred to as Boundary Condition Enforcing (BCE) markers which are used to impose no-slip and impenetrability conditions. Such BCE markers are associated both with the solid suspended particles and with any confining boundary walls and are distributed in a narrow layer on and below the surface of solid objects. The ensuing fluid–solid interaction forces are mapped into generalized forces on the rigid and flexible bodies and subsequently used to update the dynamics of the solid objects according to rigid body motion or ANCF method. The robustness and performance of the simulation algorithm is demonstrated through several numerical simulation studies.

A. Pazouki (✉) · R. Serban · D. Negrut
University of Wisconsin-Madison, 2035 Mechanical Engineering Building,
1513 University Avenue, Madison, WI 53706, USA
e-mail: pazouki@wisc.edu

R. Serban
e-mail: serban@wisc.edu

D. Negrut
e-mail: negrut@wisc.edu

2.1 Introduction

Engineers commonly rely on prototypes and physical testing when performing design and analysis tasks. Unfortunately, such work can be expensive and time consuming. Because computational hardware continues to advance in terms of both processing speed and memory size, a trend is growing in which computer simulation is used to augment and, in some cases, replace large amounts of experimental work. With increasing computational power, engineers are able to perform faster, larger, and more accurate simulations. Computer simulation has several advantages over physical experiments. Through simulation, engineers may study a range of parameter values that would prove too costly or too dangerous to study experimentally. Moreover, computer simulation can produce representative data that experimental measurements could never achieve. Experimental insights are limited by the position, fidelity, and number of sensors, whereas a simulation inherently tracks the state of every component of the system. For example, simulation can generate, in a non-intrusive fashion, the set of forces acting between all the individual bodies in a flow of suspension.

Current simulation capabilities are sometimes inadequate to capture phenomena of interest. This problem is especially evident when simulating the dynamics of Fluid–Solid Interaction (FSI) systems, which may contain tens of thousands of rigid and deformable bodies that interact directly or through the fluid media. The ability to solve such large problems will require significant improvements in terms of both algorithms and implementation.

To alleviate computational limitations, numerical simulation approaches devised for the general category of FSI problems usually suppress some physics depending on the specific application. For instance, several approaches have been proposed to study characteristics of the flow of particle suspension. These include Eulerian–Eulerian (EE) approaches, where the solid phase is considered as a continuum [14, 16, 46]; Lagrangian particle tracking, also known as Lagrangian Numerical Simulation (LNS) approaches, which either consider a one-way coupling of fluid and solid phase, or else introduce a collective momentum exchange term to the fluid equation [2, 30]; Eulerian–Lagrangian (EL) approaches, where the Lagrangian solid phase moves with/within the Eulerian grid used for fluid simulation [17, 21, 25]; and Lagrangian–Lagrangian (LL) approaches, where both phases are modeled within a Lagrangian framework [36, 38, 39]. As in EE methodologies, LNS approaches rely on empirical forms of hydrodynamic fluid–solid forces, determined mostly for dilute conditions where the particle–particle interaction is neglected.

Similar approaches are also applied to the fluid–structure interaction. In this document, the focus is primarily on the LL approaches, particularly those geared towards large deformation favored by the multibody dynamics community (some studies on problems involving small structural deformation using a Lagrangian representation of fluid flow are provided in [1, 4, 28]).

The body of work on FSI problems using Lagrangian fluid representation and large structural deformation is very limited. Schörgenhuber et al. [42] presented

a co-simulation approach for the FSI problems. In their approach, they used a heuristic force field for the coupling of the fluid and flexible objects, modeled via Smoothed Particle Hydrodynamics (SPH) and Absolute Nodal Coordinate formulation (ANCF), respectively. The suggested force field, which involves some heuristic parameters to enforce the fluid–solid coupling, cannot approximate the FSI interaction at a resolution finer than that of the fluid discretization. In this sense, it is equivalent to all other approaches proposed for the implementation of wall boundary condition with the caveats that: (1) the procedure of finding the minimum distance between fluid markers and solid surfaces can be prohibitively tedious, particularly for complex shapes; and (2) the wrong choice of heuristic parameters at a certain flow condition can result in either an inexact coupling or a stiff force model which can lead to numerical instability. Additionally, little, if any, is said about the performance of the co-simulation approach. Similarly, Hu et al. [22] approached the FSI problem using SPH and ANCF; however, they implemented the method of moving boundary to couple the fluid dynamics to solid objects.

This contribution is a further development to FSI simulation approaches presented in [22, 38, 39, 42] and also includes a moving boundary approach for two-way fluid–solid coupling implemented through the use of so-called Boundary Condition Enforcing (BCE) markers. Neither Schörghamer et al. [42] nor Hu et al. [22] addressed the solid–solid interaction required for many-body FSI problems. In the present work, support for many-body FSI problems, such as those encountered in suspension and polymer flow, is provided by incorporating a lubrication force model. In addition, we have continued our previous validation efforts by benchmarking the dynamics of flexible bodies against that of rigid objects, a study which links the validation of flexible bodies to that of rigid bodies presented in [39]. Finally, we provide a high performance implementation that leverages parallel computing on Graphical Processing Unit (GPU) cards. A complete scaling and time analysis performed herein demonstrate a typical ten-fold speedup compared to the results provided in [22] for problems of comparable size.

The remainder of this document is organized as follows. The various algorithmic components of the proposed simulation framework are discussed in Sect. 2.2, with details on their high performance computing implementation provided in Sect. 2.3. We provide simulation results in Sect. 2.4, including validation and parametric studies, and conclude with some final remarks in Sect. 2.5.

2.2 Simulation Methodology

The simulation framework developed herein relies on: (i) SPH for the simulation of fluid flow, (ii) Newton–Euler 3D rigid body equations of motion, and (iii) ANCF to capture the dynamics of deformable objects. The remainder of this section describes in more details each of these algorithmic components, including a discussion on the formulation adopted for fluid–solid interaction through BCE markers in Sect. 2.2.4

and the methodology used for short range solid–solid interaction through a lubrication force model in Sect. 2.2.5.

2.2.1 The Smoothed Particle Hydrodynamics Method

SPH Liu and Liu [29], Monaghan [31, 34] is a Lagrangian method that probes the fluid domain at a set of moving markers. Each marker has an associated kernel function $W(\mathbf{r}, h)$ defined over a support domain $S(h)$, where \mathbf{r} is the distance from the SPH marker and h is a characteristic length that defines the kernel smoothness. The kernel function should converge to the Dirac delta function as the size of the support domain tends to zero: $\lim_{h \rightarrow 0} W(\mathbf{r}, h) = \delta(\mathbf{r})$, be symmetric: $W(\mathbf{r}, h) = W(-\mathbf{r}, h)$, and normal: $\int_S W(\mathbf{r}, h) dV = 1$, where dV denote the differential volume. Based on the aforementioned properties, an SPH spatial discretization results in a second order numerical method. Kernel functions must satisfy additional properties [29]; most importantly, they should be positive and monotonically decreasing functions of \mathbf{r} . In addition, for computational efficiency, it is advantageous to only consider kernel functions with compact support. A typical kernel function, used throughout this work, is the standard cubic spline kernel, defined as:

$$W(q, h) = \frac{1}{4\pi h^3} \times \begin{cases} (2 - q)^3 - 4(1 - q)^3, & 0 \leq q < 1 \\ (2 - q)^3, & 1 \leq q < 2, \\ 0, & q \geq 2 \end{cases} \quad (2.1)$$

where $q = |\mathbf{r}|/h$. In general, the radius of the support domain, κh (see Fig. 2.1), is proportional to the characteristic length h , with $\kappa = 2$ for the kernel function of Eq. (2.1).

With ρ and μ denoting the fluid density and viscosity, \mathbf{v} and p the flow velocity and pressure, and m the mass associated with an SPH marker, the continuity equation

$$\frac{d\rho}{dt} = -\rho \nabla \cdot \mathbf{v}, \quad (2.2)$$

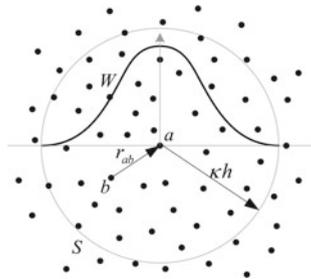
and the momentum equation

$$\frac{d\mathbf{v}}{dt} = -\frac{1}{\rho} \nabla p + \frac{\mu}{\rho} \nabla^2 \mathbf{v} + \mathbf{f}, \quad (2.3)$$

are discretized within the SPH framework as [35]:

$$\frac{d\rho_a}{dt} = \rho_a \sum_b \frac{m_b}{\rho_b} (\mathbf{v}_a - \mathbf{v}_b) \cdot \nabla_a W_{ab}, \quad (2.4)$$

Fig. 2.1 Illustration of the kernel, W , and support domain, S . SPH markers are shown as *black dots*. For 2D problems the support domain is a *circle*, while for 3D problems it is a *sphere*



and

$$\frac{d\mathbf{v}_a}{dt} = - \sum_b m_b \left(\left(\frac{p_a}{\rho_a^2} + \frac{p_b}{\rho_b^2} \right) \nabla_a W_{ab} + \Pi_{ab} \right) + \mathbf{f}_a. \quad (2.5)$$

In Eq. (2.5), indices a and b denote the SPH markers, as shown in Fig. 2.1, and

$$\Pi_{ab} = - \frac{(\mu_a + \mu_b) \mathbf{r}_{ab} \cdot \nabla_a W_{ab}}{\bar{\rho}_{ab}^2 (r_{ab}^2 + \varepsilon \bar{h}_{ab}^2)} \mathbf{v}_{ab} \quad (2.6)$$

imposes the viscous force based on the discretization of the ∇^2 operator, where ε is a regularization coefficient. Here ∇_a indicates the gradient with respect to \mathbf{r}_a , i.e. $\partial/\partial\mathbf{r}_a$. Quantities with an over-bar are the average of the corresponding quantities for markers a and b . Summations in the above equations are over all markers within the support domain of marker a . We have evaluated several definitions for the viscosity, as well as different discretizations of ∇^2 [34, 35] in conjunction with simulation of transient Poiseuille flow and concluded that Π_{ab} of Eq. (2.6) leads to the most accurate results for the widest range of Reynolds numbers. It is also worth noting that Eq. (2.6) makes use of the physical fluid viscosity, unlike the use of tuning parameters in artificial viscosity formulations [34].

The pressure p is evaluated using an equation of state [34]:

$$p = \frac{c_s^2 \rho_0}{\gamma} \left\{ \left(\frac{\rho}{\rho_0} \right)^\gamma - 1 \right\}, \quad (2.7)$$

where ρ_0 is the fluid reference density, γ is a parameter controlling the stiffness of the pressure-density relationship, and c_s is the speed of sound. In the weakly compressible SPH method, c_s is adjusted based on the maximum speed of the flow, V_{\max} , to keep the flow compressibility below any arbitrary value. We chose $\gamma = 7$ and $c_s = 10 \cdot V_{\max}$, which allows 1 % flow compressibility [34]. The fluid flow Eqs. (2.4) and (2.5) are solved in conjunction with the kinematic equation

$$\frac{d\mathbf{r}_a}{dt} = \mathbf{v}_a \quad (2.8)$$

to update the positions of all SPH markers.

Compared to Eq. (2.4), which evaluates the time derivative of the density, the original SPH summation formula calculates the density according to

$$\rho_a = \sum_b m_b W_{ab}. \quad (2.9)$$

Equation (2.4) was preferred to Eq. (2.9) since it produced a smooth density field and worked well for markers close to the boundaries, namely free surfaces, solid interfaces, and wall boundaries. However, Eq. (2.4) does not guarantee consistency between density at a marker and the associated mass and volume [6, 33, 35]. On the other hand, using Eq. (2.9) has problems of its own, in particular large variations in the density field, especially close to the boundary. One of the approaches suggested to resolve this issue is to combine the two methods in a so-called ‘‘density re-initialization technique’’ [9] in which Eq. (2.4) is enforced at each time step while Eq. (2.9) is used to correct any mass-density inconsistencies every n time steps. The results reported herein were obtained with $n = 10$. The Moving Least Squares method or a normalized version of Eq. (2.9) are alternative solutions to the aforementioned issues [9, 11].

Finally, to prevent extensive overlap of marker support domains and enhance incompressibility of the flow, we employ the extended SPH approach (XSPH) as described in [32]. The XSPH correction takes into account the velocity of neighboring markers through a mean velocity evaluated within the support of a nominal marker a as

$$\langle \mathbf{v}_a \rangle = \mathbf{v}_a + \Delta \mathbf{v}_a, \quad (2.10)$$

where

$$\Delta \mathbf{v}_a = \zeta \sum_b \frac{m_b}{\bar{\rho}_{ab}} (\mathbf{v}_b - \mathbf{v}_a) W_{ab} \quad (2.11)$$

and $0 \leq \zeta \leq 1$ adjusts the contribution of velocities of neighboring markers. All simulations presented in this work were obtained with $\zeta = 0.5$. The modified velocity calculated from Eq. (2.10) replaces the original velocity in the density and position update equations, but not in the momentum equation [9].

2.2.2 Rigid Body Dynamics

The dynamics of rigid bodies is fully characterized by the Newton–Euler equations of motion (EOM), see for instance [18]. For each body $i = 1, 2, \dots, n_b$ present in the system, we have:

$$\frac{d\mathbf{V}_i}{dt} = \frac{\mathbf{F}_i}{M_i}, \quad (2.12)$$

$$\frac{d\mathbf{X}_i}{dt} = \mathbf{V}_i, \quad (2.13)$$

$$\frac{d\boldsymbol{\omega}'_i}{dt} = \mathbf{J}'_i{}^{-1} \left(\mathbf{T}'_i - \tilde{\boldsymbol{\omega}}'_i \mathbf{J}'_i \boldsymbol{\omega}'_i \right), \quad (2.14)$$

$$\frac{d\mathbf{q}_i}{dt} = \frac{1}{2} \mathbf{G}_i^T \boldsymbol{\omega}'_i, \quad (2.15)$$

and

$$\mathbf{q}_i^T \mathbf{q}_i - 1 = 0, \quad (2.16)$$

where \mathbf{F}_i and \mathbf{T}'_i represent the external forces and torques acting on body i , including fluid–solid interaction forces obtained as described in Sect. 2.2.4. The quantities $\mathbf{X}_i \in \mathbb{R}^3$ and $\mathbf{q}_i \in \mathbb{R}^4$ denote the position vector and rotation quaternion, while \mathbf{V}_i , $\boldsymbol{\omega}'_i \in \mathbb{R}^3$ represent the linear and angular body velocities. The mass and moment of inertia are denoted by M_i and \mathbf{J}'_i , respectively. Quantities with a prime symbol are represented in the rigid body local reference frame. Given $\mathbf{a} = [a_x, a_y, a_z]^T \in \mathbb{R}^3$ and $\mathbf{q} = [q_x, q_y, q_z, q_w]^T \in \mathbb{R}^4$, the auxiliary matrices $\tilde{\mathbf{a}}$ and \mathbf{G} are defined as:

$$\tilde{\mathbf{a}} = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix} \quad \text{and} \quad \mathbf{G} = \begin{bmatrix} -q_y & q_x & q_w & -q_z \\ -q_z & -q_w & q_x & q_y \\ -q_w & q_z & -q_y & q_x \end{bmatrix}. \quad (2.17)$$

2.2.3 Flexible Body Dynamics

For the simulation of flexible solid bodies suspended in the fluid, we adopt the ANCF formulation [44] which allows for large deformations and large rigid body rotations. While extension to other elastic elements is straightforward, in the current Chrono::Fluid implementation we only support gradient deficient ANCF beam elements which are used to model slender flexible bodies composed of n_e adjacent ANCF beam elements. In this approach, we model the flexible bodies using a number $n_n = n_e + 1$ of equally-spaced node beam elements, each represented by 6 coordinates, $\mathbf{e}_j = [\mathbf{r}_j^T, \mathbf{r}_{j,x}^T]^T$, $j = 0, 1, \dots, n_e$, representing the three components of the global position vector of the node and the three components of the position vector gradient. This is therefore equivalent to a model using n_e ANCF beam elements with $6 \times n_n$ continuity constraints, but is more efficient in that it uses a minimal set of coordinates. We note that formulations using gradient deficient ANCF beam elements display no shear locking problems [15, 43, 45] and, due to the reduced number

of nodal coordinates, are more efficient than fully parameterized ANCF elements. However, gradient deficient ANCF beam elements cannot describe a rotation about its axis and therefore cannot model torsional effects.

Consider first a single ANCF beam element of length ℓ . The global position vector of an arbitrary point on the beam centerline, specified through its element spatial coordinate $0 \leq x \leq \ell$, is then obtained as

$$\mathbf{r}(x, \mathbf{e}) = \mathbf{S}(x)\mathbf{e}, \quad (2.18)$$

where $\mathbf{e} = [\mathbf{e}_l^T, \mathbf{e}_r^T]^T \in \mathbb{R}^{12}$ is the vector of element nodal coordinates. With \mathbf{I} being the 3×3 identity matrix, the 3×12 shape function matrix $\mathbf{S} = [S_1\mathbf{I} \ S_2\mathbf{I} \ S_3\mathbf{I} \ S_4\mathbf{I}]$ is defined using the shape functions [44]

$$\begin{aligned} S_1 &= 1 - 3\xi^2 + 2\xi^3 \\ S_2 &= \ell (\xi - 2\xi^2 + \xi^3) \\ S_3 &= 3\xi^2 - 2\xi^3 \\ S_4 &= \ell (-\xi^2 + \xi^3), \end{aligned} \quad (2.19)$$

where $\xi = x/\ell \in [0, 1]$.

The element EOM are then written as

$$\mathbf{M}\ddot{\mathbf{e}} + \mathbf{Q}^e = \mathbf{Q}^a, \quad (2.20)$$

where \mathbf{Q}^e and \mathbf{Q}^a are the generalized element elastic and applied forces, respectively, and $\mathbf{M} \in \mathbb{R}^{12 \times 12}$ is the symmetric consistent element mass matrix defined as

$$\mathbf{M} = \int_{\ell} \rho_s A \mathbf{S}^T \mathbf{S} dx. \quad (2.21)$$

The generalized element elastic forces are obtained from the strain energy expression [44] as

$$\mathbf{Q}^e = \int_{\ell} EA \varepsilon_{11} \left(\frac{\partial \varepsilon_{11}}{\partial \mathbf{e}} \right)^T dx + \int_{\ell} EI \kappa \left(\frac{\partial \kappa}{\partial \mathbf{e}} \right)^T dx, \quad (2.22)$$

where $\varepsilon_{11} = (\mathbf{r}_x^T \mathbf{r}_x - 1)/2$ is the axial strain and $\kappa = \|\mathbf{r}_x \times \mathbf{r}_{xx}\|/\|\mathbf{r}_x\|^3$ is the magnitude of the curvature vector. The required derivatives of the position vector \mathbf{r} can be easily obtained from Eq. (2.18) in terms of the derivatives of the shape functions as $\mathbf{r}_x(x, \mathbf{e}) = \mathbf{S}_x(x)\mathbf{e}$ and $\mathbf{r}_{xx}(x, \mathbf{e}) = \mathbf{S}_{xx}(x)\mathbf{e}$.

External applied forces, in particular the forces due to the interaction with the fluid (see Sect. 2.2.4), are included as concentrated forces at a BCE marker. The corresponding generalized forces are obtained from the expression of the virtual work as

$$\mathbf{Q}^a = \mathbf{S}^T(x_a)\mathbf{F}, \quad (2.23)$$

where \mathbf{F} is the external point force and the shape function matrix is evaluated at the projection onto the element's centerline of the force application point. If considered, the generalized gravitational force can be computed as

$$\mathbf{Q}^g = \int_{\ell} \rho_s A \mathbf{S}^T \mathbf{g} dx. \quad (2.24)$$

In the above expressions, ρ_s represents the element mass density, A is the cross section area, E is the modulus of elasticity, and I is the second moment of area.

The EOM for a slender flexible body composed of n_e ANCF beam elements are obtained by assembling the elemental EOMs of Eq. (2.20) and taking into consideration that adjacent beam elements share 6 nodal coordinates. Let $\hat{\mathbf{e}} = [\mathbf{e}_0^T, \mathbf{e}_1^T, \dots, \mathbf{e}_{n_e}^T]^T$ be the set of independent nodal coordinates; then the nodal coordinates for the j th element can be written using the mapping

$$\begin{bmatrix} \mathbf{e}_l \\ \mathbf{e}_r \end{bmatrix}_j = \mathbf{B}_j \hat{\mathbf{e}}, \quad \text{with } \mathbf{B}_j = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \dots & \mathbf{I}_3 & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{I}_3 & \dots & \mathbf{0} \end{bmatrix} \quad (2.25)$$

and the assembled EOMs are obtained, from the principle of virtual work, as follows. Denoting by \mathbf{M}_j be the element mass matrix of Eq. (2.21) for the j th ANCF beam element, it can be written in block form as

$$\mathbf{M}_j = \begin{bmatrix} \mathbf{M}_{j,ll} & \mathbf{M}_{j,lr} \\ \mathbf{M}_{j,rl} & \mathbf{M}_{j,rr} \end{bmatrix}, \quad (2.26)$$

where $\mathbf{M}_{j,lr} = \mathbf{M}_{j,rl}^T$ and all sub-blocks have dimension 6×6 . Here, l denotes the *left* end of the beam element, i.e., the node characterized by the nodal coordinates \mathbf{e}_{j-1} , while r corresponds to the node with coordinates \mathbf{e}_j . With a similar decomposition of a generalized element force into

$$\mathbf{Q}_j = \begin{bmatrix} \mathbf{Q}_{j,l} \\ \mathbf{Q}_{j,r} \end{bmatrix} \quad (2.27)$$

we obtain

$$\hat{\mathbf{M}}\ddot{\mathbf{e}} = \hat{\mathbf{Q}}^a - \hat{\mathbf{Q}}^e \quad (2.28)$$

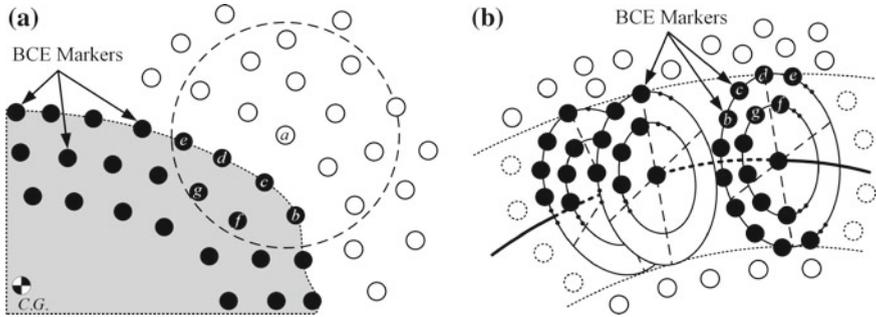


Fig. 2.2 Fluid–solid interaction using BCE markers attached to a body: (a) rigid body; (b) flexible beam. BCE and fluid markers are represented by *black* and *white circles*, respectively. The BCE markers positioned in the interior of the body should be placed to a depth no larger than the size of the compact support S of the kernel W

2.2.5 Short Range Interaction

Dry friction models typically used to characterize the dynamics of granular materials [3, 23, 24] do not resolve the impact of solid surfaces in hydrodynamics media. In practice, it is unfeasible to resolve the short-range, high-intensity impact forces in wet media due to the computational limits on space and time resolution. In reality, particle boundaries are not smooth and physical contact can happen [20]. By assuming smooth surfaces, Davis et al. followed the Hertz contact theory of linear elasticity to calculate the pressure at the interface of two approaching elastic spheres in close proximity [10]. Their calculation showed that particles do not rebound at small Stokes number, $St = (2/9)(\rho_s/\rho)Re_p$, where ρ_s and Re_p are the solid particle density and particle Reynolds number, respectively. The minimum St for a rebound after the hydroelastic impact depends on the spheres' rigidity. For rigid spheres, rebound happens at $St > 10$. An alternative approach to calculate the singular forces at contact relies on lubrication theory [13]. Ladd [26] proposed a normal lubrication force between two spheres that increases rapidly as the distance between particles approaches zero and therefore prevents the actual touching of the spheres:

$$\mathbf{F}_{ij}^{lub} = \min \left\{ -6\pi\mu a_{ij}^2 \left(\frac{1}{s} - \frac{1}{\Delta_c} \right), 0 \right\} \cdot \mathbf{v}_{n_{ij}}, \quad \text{where } \frac{1}{a_{ij}} = \frac{1}{a_i} + \frac{1}{a_j}. \quad (2.31)$$

Here, a_i and a_j are the sphere radii, $\mathbf{v}_{n_{ij}}$ is the normal component of the relative velocity, and s is the distance between surfaces. For $s > \Delta_c$, $\mathbf{F}_{ij}^{lub} = 0$ and the spheres are subject only to hydrodynamic forces. Ladd and Verberg [27] demonstrated good agreement of the proposed lubrication force with Brenner's exact solution [7].

Equation (2.31) provides a simplistic model for the estimation of the lubrication force in normal direction. Generalization of this model to non-spherical objects requires the calculation of the minimum distance and curvature of the

contact surfaces. By adopting the approach proposed in [12] for lubrication force in lattice Boltzmann method, we calculate the partial lubrication force by modifying Eq. (2.31) as

$$\mathbf{F}_{ij}^{lub} = \sum_k \mathbf{f}_{ij}^k, \quad \text{with } \mathbf{f}_{ij}^k = \min \left\{ -\frac{3}{2}\pi\mu h^2 \left(\frac{1}{s^*} - \frac{1}{\Delta_c} \right), 0 \right\} \cdot \mathbf{v}_{n_{ij}}^*, \quad (2.32)$$

where s^* and $\mathbf{v}_{n_{ij}}^*$ denote the markers relative distance and velocity, respectively, and the summation is over all interacting markers of two solid objects.

2.3 GPU-Based Implementation

Chrono::Fluid [8], an open-source simulation framework for fluid–solid interaction, relies on a second order explicit Runge-Kutta method [5] for time integration of fluid, rigid, and flexible bodies, and a parallel implementation of the spatial subdivision method on the GPU for construction of the markers neighbor lists. In what follows, the computation kernels and their implementations are described with more details.

At the beginning of each time step, a neighbor list is assembled to indicate the set of markers that fall within the kernel support of each marker; if N markers are used in the simulation, N lists are generated. The force components appearing on the right hand side of Eqs. (2.4), (2.5), and (2.31) are subsequently computed based on these neighbor lists. Two different functions are called to capture the interaction between markers according to their types; i.e., fluid or solid, via SPH or the short range interaction model described in Sect. 2.2.5. In the second stage, the state of the fluid markers, including position, velocity, and density, is updated based on Eqs. (2.4), (2.5), and (2.8). The state of each rigid body is updated according to Eqs. (2.12) through (2.15). This is followed by time integration of deformable body motion according to Eq. (2.28). Since a rigid wall boundary is a particular instance of a rigid body (with zero or other predefined velocity), it requires no special treatment.

Stable integration of the SPH fluid equations requires step-sizes which are also appropriate for propagating the dynamics of any rigid solids in the FSI system. However, integration of the dynamics of deformable bodies, especially as their stiffness increases, may require smaller time steps. To accommodate this requirement, while minimizing any adverse effects on the overall simulation efficiency, we have implemented a simple dual-rate integration scheme using intermediate steps for the integration of the flexible dynamics EOMs (typically $\Delta t_{SPH}/\Delta t_{ANCF} = 10$, although stiffer problems may require ratios of up to 50). We note that typical FSI simulation models involve a number of SPH markers many orders of magnitude larger than that of ANCF nodal coordinates required for the flexible bodies. As such, the execution time required for integration of the flexible body dynamics is a negligible fraction of the computation time for propagating the SPH equations and therefore the dual-rate integration scheme has no effect on the net overall simulation efficiency.

The above algorithm was implemented to execute in parallel on GPU cards using CUDA [37]. The hardware used to run the simulations that produced the results reported in this contribution, NVIDIA Kepler K20X, has 2688 parallel scalar processors. At each time step, five different tasks are executed on the GPU to (1) calculate the inter-marker forces, (2) carry out fluid time integration, (3) carry out rigid body time integration, (4) carry out deformable body time integration, and (5) enforce boundary conditions. The lists of neighbors needed to evaluate the inter-marker forces are generated via a proximity computation algorithm based on a decomposition of the computation domain into cubic bins. The side length of each bin is roughly equal to the size of the support domain of an SPH marker. A hash table is used to sort the markers according to their location in the domain. Based on the sorted hash table, each marker accesses the list of markers intersecting its own and neighboring bins to calculate the forcing terms. The proximity computation algorithm uses the parallel sorting and scan algorithms provided by the Thrust library [19].

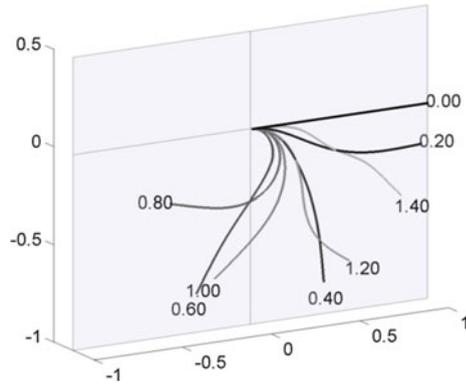
To improve the code vectorization through coalesced memory access and use of fast memory (L1/L2 cache, shared memory, and registers), each computation task was implemented as a sequence of light-weight GPU kernels. For instance, different computation kernels are implemented to update the attributes of the solid bodies, including force, moment, rotation, translation, linear and angular velocity, and location of the BCE markers. A similar coding style was maintained for the density re-initialization, boundary condition implementation, and mapping of the markers' data on an Eulerian grid for post processing.

2.4 Results and Discussion

The robustness and accuracy of the fluid flow and coupled fluid-rigid body simulation was demonstrated in previous work. See [39] for a comprehensive set of validation studies of rigid particle migration and suspension distribution in pipe flow. Herein, we focus on recent extensions to Chrono::Fluid to support fluid-deformable body interaction and present additional numerical experiments to validate the flexible body simulation algorithm, as well as several simulation-based studies involving coupling of fluid flow and deformable bodies.

The simulations presented in this section involve relatively soft beams (with a modulus of elasticity $E \leq 20$ MPa) that are either unconstrained or else anchored at one end. Since computational efficiency of the FSI code is directly related to the number of nodal coordinates used to model the flexible beams, we first conducted a parametric study to identify the minimum number of ANCF beam elements required to accurately capture the dynamics of interest in the subsequent experiments. In this set of experiments, we considered a cantilever of length $L = 1$ m and diameter $d = 0.04$ m with density $\rho_s = 7,200$ kg/m³ and modulus of elasticity $E = 20$ Mpa under gravity ($g = -9.81$ m/s²) in vacuum or immersed in fluid of various viscosities. Simulation results using different number of ANCF beam elements ($n_e = 2, 3, 4, 5$) showed acceptable convergence at all discretizations and virtually identical results

Fig. 2.3 Time snapshots of a flexible cantilever moving under the action of gravity (in vacuum). The *darker colors* denote earlier stages of the motion



for $n_e \geq 4$. Figure 2.3 shows a few time snapshots from a dynamic simulation of a cantilever modeled with $n_e = 4$ ANCF beam elements, the value which was selected for all subsequent simulations.

2.4.1 Floating Beam in Poiseuille Flow

Ongoing work is aimed at validating the fluid-deformable solid interaction code against experimental and analytical results [40]. Here we present a comparison against the already validated fluid-rigid solid simulation code. For this purpose we conducted a series of numerical experiments involving short stiff deformable beams and equivalent rigid cylinders free floating in channel Poiseuille flow.

The validation test was performed using a straight beam with $L = 0.2$ m, $\rho_s = 7,200$ kg/m³, $E = 20$ MPa, $d = 0.04$ m and a rigid cylinder with the same density and geometry. The beam and rigid cylinder were subjected to an accelerating channel flow aligned with the global x axis with final steady state Reynolds number $Re_c = \rho V_{ave} w / \mu = 100$, where $\rho = 1,000$ kg/m³, $\mu = 1$ N s/m², average velocity $V_{ave} = 0.2$ m/s, and channel width $w = 1$ m. The beam and cylinder were initially perpendicular to the flow and rotated in the yz plane. Comparisons of the resulting beam orientation angles, relative to the global x , y , and z axes, and of the time evolution of the velocity in the x direction of the beam center velocity are presented in Fig. 2.4. The results show good agreement with differences due to the inability of the gradient deficient ANCF beam element model to capture rotation about the beam's axis.

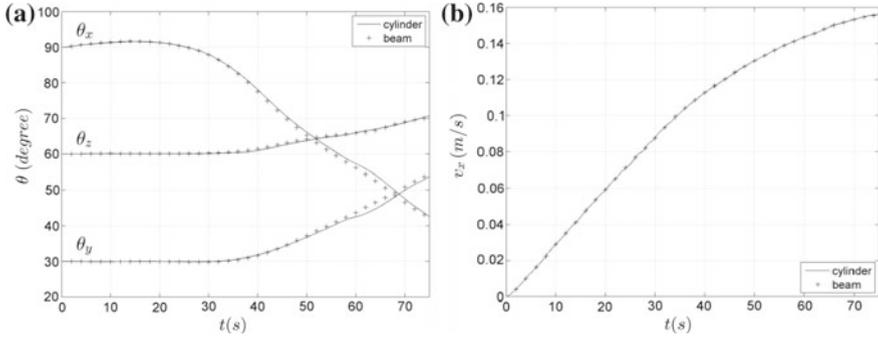


Fig. 2.4 Comparison of the dynamics of a rigid cylinder and of a corresponding stiff deformable beam under accelerating channel flow: **(a)** beam orientation; **(b)** center velocity

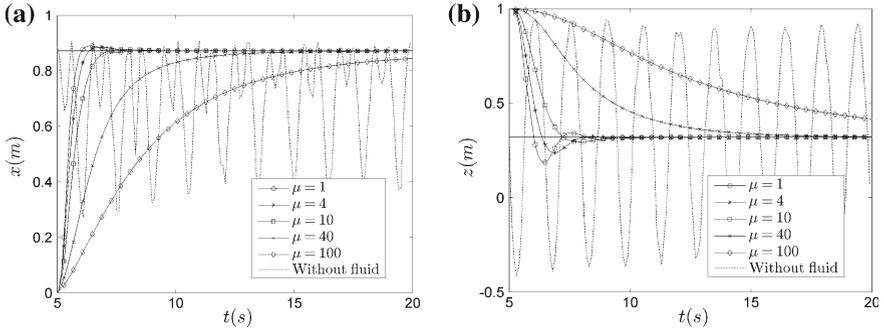


Fig. 2.5 Motion of a cantilever beam in fluid of different viscosities: **(a)** tip displacement in x direction; **(b)** tip displacement in z direction

2.4.2 Flexible Cantilever Immersed in Fluid: Effect of Viscosity

Through a parametric study of the motion of a cantilever moving under the action of gravity in viscous fluid, we investigated the effect of viscosity on the motion of the beam’s tip. As shown in Fig. 2.5, the beam motion switches from oscillatory to critically damped motion as the viscosity increases. For the beam parameters used in this study, namely $L = 1$ m, $d = 0.04$ m, $\rho_s = 7,200$ kg/m³, and $E = 20$ MPa, the switch between the two behaviors is observed to occur around $\mu \simeq 10$ N s/m². It was also noticed that viscosity has little effect on the trajectory of the beam tip (plots are not provided). Nevertheless, compared to the case of a cantilever moving in vacuum, when immersed in fluid, the tip moves on a much shorter path. This deviation, i.e. having the same trajectory regardless of the fluid viscosity, which is different from that of a cantilever in vacuum, is most probably due to the pressure drag which is added to the viscous drag considered herein.

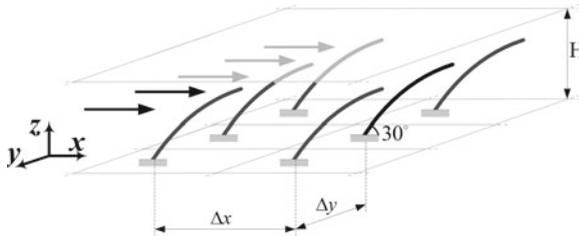


Fig. 2.6 Arrays of flexible cantilever beams in laminar channel flow. The beams, laid out in an uniform grid, are anchored at an angle of 30° in the direction of the flow

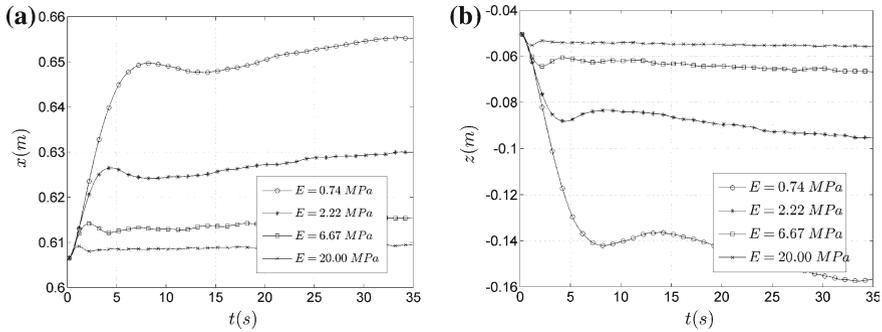


Fig. 2.7 Motion of a cantilever beam of different elasticity modulus in laminar channel flow: (a) tip displacement in x direction; (b) tip displacement in z direction

2.4.3 Impulsively Started Motion of Cantilevers in Channel Flow: Effect of Elasticity

Vibration behavior of flexible beams in viscous fluid was studied by considering an array of cantilevers in channel flow. Unlike the test described in Sect. 2.4.2, here the flexible cantilevers are initially at rest when they are hit by a laminar channel flow. This model can be used to study the effect of horizontal waves on beams submerged in a fluid.

The array of flexible cantilevers is laid out in the xy plane, with $(\Delta x, \Delta y) = (1.2, 0.4)$ m, as shown in Fig. 2.6, thus allowing interaction of the beams through the flow. Each beam is anchored in the xz plane with an angle of 30° with respect to the y axis. The fluid, with density $\rho = 1,000$ kg/m³ and viscosity $\mu = 1$ N s/m², flows in the x direction between two planes spaced by $H = 1$ m vertically.

Figure 2.7 shows the tip deformation of one cantilever beam for different modulus of elasticity in the range $E \in (0.25, 20)$ MPa. All other beam parameters were kept fixed at $L = 0.7$ m, $d = 0.04$ m, and $\rho_s = 7,200$ kg/m³.

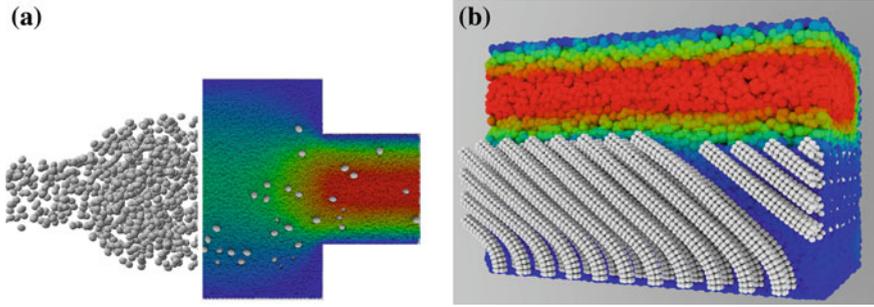


Fig. 2.8 FSI problems considered for scalability analysis: (a) flow of a dense suspension of rigid particles through a step pipe; for clarity, the *left half* of the image shows the rigid particles only, while the *right half* shows both rigid particles and SPH markers at the pipe mid-section; (b) channel flow over an array of flexible cantilever beams; for visualization purposes only, marker sizes are artificially changed

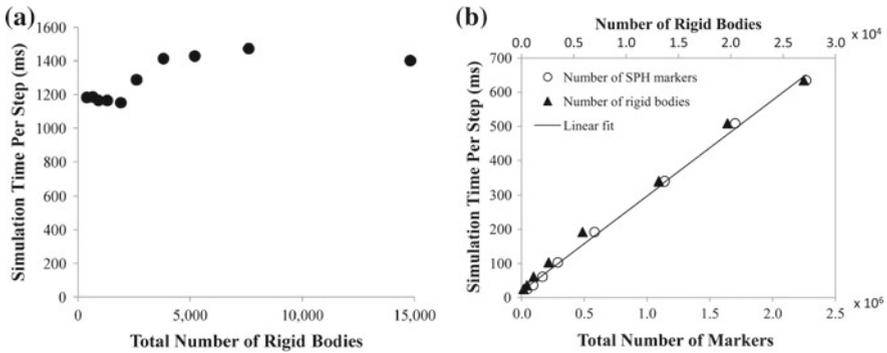


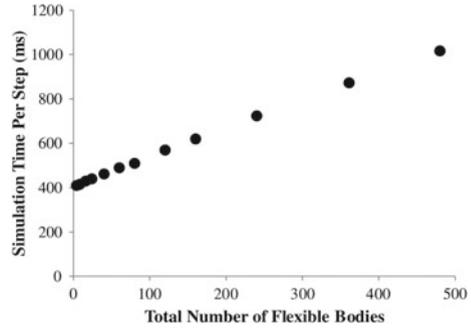
Fig. 2.9 Scaling analysis of Chrono::Fluid for fluid-rigid body interaction problems: (a) simulation time versus number of rigid bodies for a total number of 3×10^6 markers; (b) simulation time as a function of combined problem size

2.4.4 Scalability Analysis

Scalability of Chrono::Fluid was investigated through simulations of multi-component problems including the flow of flexible and rigid objects in flow, samples of which are provided in Fig. 2.8.

As shown in Fig. 2.9a, an increase in the number of rigid bodies present in the system only marginally affects the total simulation time. This is due to the fact that the number of BCE markers associated with solid bodies is only a very small fraction of the number of SPH discretization markers, the latter dictating to a very large extent the required computation time. We must however mention that, as the concentration of solid objects increases, smaller time steps are required since the probability of short-range, high-frequency interactions increases. The same conclusion can be reached

Fig. 2.10 Scaling analysis of Chrono::Fluid for fluid-flexible body interaction problems: simulation time versus number of flexible bodies for a total number of 1.5×10^6 markers



from the results presented in Fig. 2.9b which shows linear growth of the simulation time with the size of the fluid-rigid body mixture problem (i.e., the combined number of SPH markers and rigid bodies).

On the other hand, as seen in Fig. 2.10, we observe only linear scalability when rigid bodies are replaced by flexible beams. This is only a consequence of the current Chrono::Fluid implementation in which the dynamics update for flexible bodies is carried out on the CPU, thus dominating the simulation time as the problem size increases. We expect this will be rectified once this stage of the simulation is also moved to the GPU.

2.5 Conclusions and Future Work

We describe a Lagrangian–Lagrangian approach for the direct numerical simulation of two-way coupled fluid–solid interaction. Building up on previous work [39], the simulation framework Chrono::Fluid was extended beyond fluid-rigid interaction to include deformable solids. For simulations of solid bodies immersed in fluid, the proposed method employs a lubrication force model for incorporating solid–solid interaction and, in the case of deformable bodies, self-contact. We describe simulation results for free-floating flexible beams in Poiseuille flow and channel flow over a grid of flexible cantilevers, and provide parametric studies of the effect of fluid viscosity and material elasticity. These results suggest that the adopted approach has good predictive capabilities and is able to capture the dynamics of the systems under consideration. Moreover, the Lagrangian–Lagrangian formulation is amenable to efficient implementation on GPU cards as indicated by the scalability studies presented herein.

Current effort is aimed at providing a GPU-only implementation, by also parallelizing the flexible body dynamics calculations and updates, with ongoing work focused on additional validation studies using both analytical and experimental data. In addition, we plan on extending the formulation to support 2D and fully-3D deformable solids modeled with ANCF.

Finally, we note that additional examples of Chrono::Fluid simulations can be found at [41]

Acknowledgments Financial support was provided in part by the National Science Foundation (grant NSF CMMI-084044). Support for the second and third authors was provided in part by Army Research Office awards W911NF-11-0327 and W911NF-12-1-0395. NVIDIA is acknowledged for providing the GPU hardware used in generating the simulation results reported herein.

References

1. Amini Y, Emdad H, Farid M (2011) A new model to solve fluid-hypo-elastic solid interaction using the smoothed particle hydrodynamics (SPH) method. *Eur J Mech B Fluids* 30(2):184–194
2. Andrews M, O’rourke P (1996) The multiphase particle-in-cell (MP-PIC) method for dense particulate flows. *Int J Multiph Flow* 22(2):379–402
3. Anitescu M, Hart GD (2004) A constraint-stabilized time-stepping approach for rigid multibody dynamics with joints, contact and friction. *Int J Numer Meth Eng* 60(14):2335–2371
4. Antoci C, Gallati M, Sibilla S (2007) Numerical simulation of fluid-structure interaction by SPH. *Comput Struct* 85(11):879–890
5. Atkinson K (1989) *An introduction to numerical analysis*. Wiley, USA
6. Benz W (1986) Smoothed particle hydrodynamics: a review. In: *Proceedings of the NATO advanced research workshop on the numerical modelling of nonlinear stellar pulsations problems and prospects*, Les Arcs, France, 20–24 Mar. Kluwer Academic Publishers, Berlin
7. Brenner H (1961) The slow motion of a sphere through a viscous fluid towards a plane surface. *Chem Eng Sci* 16(3):242–251
8. Chrono::Fluid (2014) An open source engine for fluid. Solid interaction. <http://armanpazouki.github.io/chrono-fluid>
9. Colagrossi A, Landrini M (2003) Numerical simulation of interfacial flows by smoothed particle hydrodynamics. *J Comput Phys* 191(2):448–475
10. Davis RH, Serayssol JM, Hinch E (1986) Elastohydrodynamic collision of two spheres. *J Fluid Mech* 163:479–497
11. Dils G (1999) Moving-least-squares-particle hydrodynamics-I. Consistency and stability. *Int J Numer Methods Eng* 44(8):1115–1155
12. Ding EJ, Aidun CK (2003) Extension of the lattice-Boltzmann method for direct simulation of suspended particles near contact. *J Stat Phys* 112(3–4):685–708
13. Durlofsky L, Brady JF, Bossis G (1987) Dynamic simulation of hydrodynamically interacting particles. *J Fluid Mech* 180(1):21–49
14. Fan LS, Zhu C (2005) *Principles of gas-solid flows*. Cambridge University Press, Cambridge
15. Gerstmayr J, Shabana A (2006) Analysis of thin beams and cables using the absolute nodal co-ordinate formulation. *Nonlinear Dyn* 45(1):109–130
16. Gidaspow D (1994) *Multiphase flow and fluidization: continuum and kinetic theory descriptions*. Academic Press, Boston
17. Glowinski R, Pan T, Hesla T, Joseph D (1999) A distributed Lagrange multiplier/fictitious domain method for particulate flows. *Int J Multiph Flow* 25(5):755–794
18. Haug E (1989) *Computer aided kinematics and dynamics of mechanical systems*. Allyn and Bacon, Boston
19. Hoberock J, Bell N Thrust: C++ template library for CUDA. <http://thrust.github.com/>
20. Hocking L (1973) The effect of slip on the motion of a sphere close to a wall and of two adjacent spheres. *J Eng Math* 7(3):207–221
21. Hu H, Patankar N, Zhu M (2001) Direct numerical simulations of fluid-solid systems using the arbitrary Lagrangian–Eulerian technique. *J Comput Phys* 169(2):427–462

22. Hu W, Tian Q, Hu H (2013) Dynamic simulation of liquid-filled flexible multibody systems via absolute nodal coordinate formulation and SPH method. *Nonlinear Dyn*, pp 1–19
23. Kruggel-Emden H, Simsek E, Rickelt S, Wirtz S, Scherer V (2007) Review and extension of normal force models for the discrete element method. *Powder Technol* 171(3):157–173
24. Kruggel-Emden H, Wirtz S, Scherer V (2008) A study on tangential force laws applicable to the discrete element method (DEM) for materials with viscoelastic or plastic behavior. *Chem Eng Sci* 63(6):1523–1541
25. Ladd AJ (1994) Numerical simulations of particulate suspensions via a discretized Boltzmann equation. *J Fluid Mech* 271(1):285–339
26. Ladd AJ (1997) Sedimentation of homogeneous suspensions of non-brownian spheres. *Phys Fluids* 9:491
27. Ladd A, Verberg R (2001) Lattice-Boltzmann simulations of particle-fluid suspensions. *J Stat Phys* 104(5–6):1191–1251
28. Lee CJK, Noguchi H, Koshizuka S (2007) Fluid-shell structure interaction analysis by coupled particle and finite element method. *Comput Struct* 85(11):688–697
29. Liu M, Liu G (2010) Smoothed particle hydrodynamics (SPH): an overview and recent developments. *Arch Computat Methods Eng* 17(1):25–76
30. McLaughlin J (1994) Numerical computation of particles-turbulence interaction. *Int J Multiph Flow* 20:211–232
31. Monaghan JJ (1988) An introduction to SPH. *Comput Phys Comm* 48(1):89–96
32. Monaghan J (1989) On the problem of penetration in particle methods. *J Comput Phys* 82(1):1–15
33. Monaghan JJ (1992) Smoothed particle hydrodynamics. *Ann Rev Astron Astrophys* 30:543–574
34. Monaghan J (2005) Smoothed particle hydrodynamics. *Rep Prog Phys* 68(1):1703–1759
35. Morris J, Fox P, Zhu Y (1997) Modeling low Reynolds number incompressible flows using SPH. *J Computat Phys* 136(1):214–226
36. Negrut D, Tasora A, Mazhar H, Heyn T, Hahn P (2012) Leveraging parallel computing in multibody dynamics. *Multibody Sys Dyn* 27(1):95–117
37. NVIDIA: CUDA developer zone (2014). <https://developer.nvidia.com/cuda-downloads>
38. Pazouki A, Negrut D (2012) Direct simulation of lateral migration of buoyant particles in channel flow using GPU computing. In: Proceedings of the 32nd computers and information in engineering conference, CIE32, 12–15 Aug 2012. American Society of Mechanical Engineers, Chicago, IL, USA
39. Pazouki A, Negrut D A numerical study of the effect of particle properties on the radial distribution of suspensions in pipe flow. Submitted to *International Journal of Multiphase Flow*
40. Sader J (1998) Frequency response of cantilever beams immersed in viscous fluids with applications to the atomic force microscope. *J Appl Phys* 84(1):64–76
41. SBEL: Vimeo page (2014). <https://vimeo.com/uwsbel>
42. Schörgenhumer M, Gruber PG, Gerstmayr J (2013) Interaction of flexible multibody systems with fluids analyzed by means of smoothed particle hydrodynamics. *Multibody Sys Dyn*, pp 1–24
43. Schwab A, Meijaard J (2005) Comparison of three-dimensional flexible beam elements for dynamic analysis: finite element method and absolute nodal coordinate formulation. In: Proceedings of the ASME 2005 IDETC/CIE, Orlando, Florida
44. Shabana A (2005) Dynamics of multibody systems. Cambridge University Press, New York
45. Shabana A (1997) Flexible multibody dynamics: review of past and recent developments. *Multibody Sys Dyn* 1:339–348
46. Zhang D, Prosperetti A (1994) Averaged equations for inviscid disperse two-phase flow. *J Fluid Mech* 267:185–220



<http://www.springer.com/978-3-319-07259-3>

Multibody Dynamics
Computational Methods and Applications
Terze, Z. (Ed.)
2014, VIII, 368 p. 162 illus., Hardcover
ISBN: 978-3-319-07259-3