

Strathprints Institutional Repository

Alexiadis, Alessio and Lockerby, Duncan A. and Borg, Matthew Karl and Reese, Jason (2013) *A Laplacian-based algorithm for non-isothermal atomistic-continuum hybrid simulation of micro and nano-flows*. *Computer Methods in Applied Mechanics and Engineering*, 264. pp. 81-94. ISSN 0045-7825

Strathprints is designed to allow users to access the research output of the University of Strathclyde. Copyright © and Moral Rights for the papers on this site are retained by the individual authors and/or other copyright owners. You may not engage in further distribution of the material for any profitmaking activities or any commercial gain. You may freely distribute both the url (<http://strathprints.strath.ac.uk/>) and the content of this paper for research or study, educational, or not-for-profit purposes without prior permission or charge.

Any correspondence concerning this service should be sent to Strathprints administrator: <mailto:strathprints@strath.ac.uk>



A Laplacian-based algorithm for non-isothermal atomistic-continuum hybrid simulation of micro and nano-flows [☆]



Alessio Alexiadis ^{a,*}, Duncan A. Lockerby ^a, Matthew K. Borg ^b, Jason M. Reese ^b

^a School of Engineering, University of Warwick, Coventry CV4 7AL, United Kingdom

^b Mechanical & Aerospace Engineering, University of Strathclyde, Glasgow G1 1XJ, United Kingdom

ARTICLE INFO

Article history:

Received 17 January 2013

Received in revised form 17 April 2013

Accepted 29 May 2013

Available online 6 June 2013

Keywords:

Atomistic-continuum hybrid modelling

Molecular dynamics

Fluid dynamics

ABSTRACT

We propose a new hybrid algorithm for incompressible micro and nanoflows that applies to non-isothermal steady-state flows and does not require the calculation of the Irving–Kirkwood stress tensor or heat flux vector. The method is validated by simulating the flow in a channel under the effect of a gravity-like force with bounding walls at two different temperatures and velocities. The model shows very accurate results compared to benchmark full MD simulations. In the temperature results, in particular, the contribution of viscous dissipation is correctly evaluated.

© 2013 The Authors. Published by Elsevier B.V. All rights reserved.

1. Introduction

Current developments in micro and nanofluidics have created the need for new computational methods that can concurrently and efficiently handle different time and length scales. At these scales in fact the continuum-fluid hypothesis loses its validity and the behaviour of the fluid should be calculated, at least in theory, from the (averaged) motion of its constitutive molecules. In practice, however, most of the time the continuum formulation can still be employed to describe the overall behaviour of the fluid, but certain ‘adjustments’ must be introduced. The standard no-slip boundary conditions, for instance, cannot always be employed in microflows, while confinement in nanochannels creates anisotropies in a fluid’s density and alterations of the molecular distribution function, which, in turn, affect all the macroscopic properties of the fluid. This phenomenon has been clearly demonstrated for water in carbon nanotubes [3,4,5,6,28,29], where self-diffusivity, hydrogen bonding, freezing point, viscosity, etc. are not only very different from those of bulk water, but also non-uniformly distributed in the nanotube. If we consider, for instance, the case of transport properties (e.g. viscosity, diffusivity, and thermal conductivity), the attempt to provide a classical correlation of the type $\text{flux} = f(\text{gradient})$ (viz. flux of momentum, heat or mass as a function of, respectively, velocity, temperature or concentration gradient)

would fail because the result would depend not only on the fluid characteristics but also on the size and the geometry of the nanochannel. There are even some macroscopic cases that are difficult to treat with traditional continuum modelling and would benefit from an atomistic treatment. Complex polymeric fluids, for instance, can be strongly non-Newtonian and the stress/velocity relation cannot always be tabulated.

In all of these cases, a full molecular approach such as molecular dynamics (MD) would provide a better picture of the fluid’s behaviour. Despite the recent advances in computer hardware and software, however, molecular simulations cannot yet handle the large number of atoms involved in many micro and nanofluidic applications. In order to tackle this problem, various atomistic-continuum hybrid (ACH) models have been recently proposed (see [27] for a review). The fluid, depending on the model, is treated globally or partially as a continuum, and described by the Navier–Stokes equations; in certain specific regions, however, molecular dynamics is employed.

One practical advantage of these hybrid methods consists in the fact that, independently, both the continuum and the atomistic numerical parts have been extensively developed in the last decades, and researchers can now take advantage of the many accessible computational fluid dynamics (CFD) and molecular dynamics (MD) codes. Current research in the field therefore focuses on the coupling between the continuum and the atomistic solvers.

There are various ACH methods available in the literature (see [27] for a review), which can be classified according to the way they exchange information between the continuum and atomistic solvers. Each of these methods has specific advantages and disadvantages according to the particular application chosen. Here, we

[☆] This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

* Corresponding author.

E-mail address: a.alexiadis@warwick.ac.uk (A. Alexiadis).

focus on the approach known as the heterogeneous multiscale method (HMM) [31] or, alternatively, as point-wise coupling (PWC) [10]. These hybrid methods are particularly useful in the case of the so-called ‘type B’ problems [31], which require the determination of first-principle-based closure terms in the constitutive parts of the macroscopic momentum equation. In our case, for instance, this means that certain specific elements of the Navier–Stokes equations (e.g. shear stress) are derived directly from atomistic calculations.

The general approach of the HMM/PWC is the following [31,10,13]. Let us assume that the value of a microscopic variable \mathbf{u} (streaming velocity) can be determined by means of a microscale model (e.g. molecular dynamics). We are not interested in the microscopic details of \mathbf{u} , but rather a coarse-grained representation \mathbf{U} (the macroscopic velocity field), which comes from the solution of the Navier–Stokes equation written in the form

$$\frac{\partial \mathbf{U}}{\partial t} = -\frac{1}{\rho} \nabla \cdot \mathbf{T} + \mathbf{g}, \quad (1)$$

where t is the time, ρ is the density, \mathbf{g} is the external force per unit mass and \mathbf{T} is the momentum flux defined as

$$\mathbf{T} = \rho \mathbf{U} \mathbf{U} + p \mathbf{I} - \boldsymbol{\tau}, \quad (2)$$

where p is the pressure, \mathbf{I} the identity matrix and $\boldsymbol{\tau}$ the shear stress. In traditional continuum modelling, an empirical relationship between the shear stress and the strain rate is introduced in order to close Eq. (2) and, therefore, Eq. (1). In HMM, the microscopic model provides all the information necessary to determine \mathbf{T} . This is usually done by means of the Irving–Kirkwood (IK) equation [18])

$$\mathbf{T}(\mathbf{r}, t) = \frac{1}{V} \left[\sum_i m_i \mathbf{v}_i \mathbf{v}_i + \frac{1}{2} \sum_{ij} \mathbf{r}_{ij} \mathbf{O}_{ij} \mathbf{f}_{ij} \Big|_{\mathbf{r}_i=\mathbf{r}} \right], \quad (3)$$

where m_i is the mass of molecule i , \mathbf{r}_i the position, \mathbf{v}_i is the velocity, \mathbf{f}_{ij} is the force acting on molecule i by the j molecule and the operator \mathbf{O}_{ij} is given by

$$\mathbf{O}_{ij} \equiv 1 - \frac{1}{2!} \mathbf{r}_{ij} \cdot \frac{\partial}{\partial \mathbf{r}} + \dots + \frac{1}{n!} \left[-\mathbf{r}_{ij} \cdot \frac{\partial}{\partial \mathbf{r}} \right]^{n-1} + \dots \quad (4)$$

In practice, the calculation of the \mathbf{O}_{ij} term can be rather complicated for non-equilibrium simulations. Ren and E [31] partially simplify this task by calculating the average flux and employing a 2D modification of the Lees–Edwards shear boundary conditions. This type of computational cell is periodic and changes its shape during the simulation in such a way as to produce a specific velocity gradient in the flow. From this gradient, the momentum flux can be calculated by Eq. (3) and introduced into Eq. (1).

In this paper, however, we prefer the simpler ‘framed’ cell employed by Hadjiconstantinou and Patera [16], where the shear stress is generated by constraining the velocity in a ‘frame’ rather than by modifying the shape of the box. The framed cell is periodic, but we cannot simply calculate the average stress in the whole box because the presence of an external buffer would produce spurious results. We need the local stress in the core region, but this complicates the \mathbf{O}_{ij} term in Eq. (3). There are other methods to calculate the stress tensor such as the method of planes [32], the volume-average approach [26,14], or the method derived from the Schweitz virial relation [25], but, in general, we must choose between a complicated computational cell (i.e. Lees–Edwards cell) and simplifying the calculation of the momentum flux, or a simple cell (i.e. framed cell) and complicating the calculation of the momentum flux. The new method we propose here does not need the direct calculation of the flux, so it avoids this issue altogether: we can use the framed cell and, at the same time, avoid the calculation of the IK equation.

Our approach, furthermore, works for steady-state systems. In one of the first hybrid methods proposed for the case of dense

fluids, the Domain Decomposition Method (DDM), the transient case was more challenging than the steady-state because the continuum time-step was partially related to the atomistic one (see [27] for details). In its original formulation [31], however, the HMM cannot be used for steady-state flows, since, if in Eq. (1) the time derivative vanishes, \mathbf{U} disappears and the continuum solution cannot be calculated. Steady-state solutions, of course, can always be obtained as limiting cases of a time-implicit simulation, but this option is not optimal when long transient periods are involved. This circumstance can limit the applicability of the method, since many micro and nanofluidic practical applications work at steady state [1].

The other benefit of the method we propose here is that it can be easily extended to non-isothermal flows and, in this article, momentum and heat transfer are actually handled together. Before hybrid models can express their full potential for engineering applications, they must be able to manage the whole spectrum of transport phenomena (i.e. momentum, heat and mass). While there is one study including heat transport in the DDM [24], the HMM is, for the moment, limited only to momentum transfer (diffusive mass transfer is still missing in both formulations). We must also consider that, at the molecular scale, heat and momentum transport are connected, which usually does not emerge in macroscopic flows at low Mach number. Isothermality, in fact, is a common assumption in engineering flows where heat is neither generated (e.g. from nuclear or chemical reactions) nor externally introduced (e.g. walls at a different temperature from the fluid). At the microscale and especially at the nanoscale, however, the relative importance of viscous dissipation increases [21] and, therefore, a velocity gradient can result in a non-negligible amount of internally generated heat. Momentum and heat transport are consequently coupled and must be solved together.

This paper is organized as follows. First, we give a general overview of how the atomistic-continuum hybrid algorithm works. Second, we describe the MD box we use for our simulation and how velocity and temperature are constrained in order to be consistent with the continuum solution. Third, we reformulate the coupling in such a way that the calculation of the IK fluxes is avoided. Finally, we validate the method against a benchmark case: non-isothermal channel flow.

2. Algorithm’s overview

Hybrid models for fluid simulations such as HMM or PWC generally work as illustrated in Fig. 1. The cyclic scheme is composed of the following four steps, which are repeated during the calculation until convergence. (Here we assume that the ‘location’ of every MD simulation corresponds to one node on the continuum grid.)

Step 1. Macroscopic solver: the continuum solution is calculated at the N discretization nodes of the macroscopic domain.

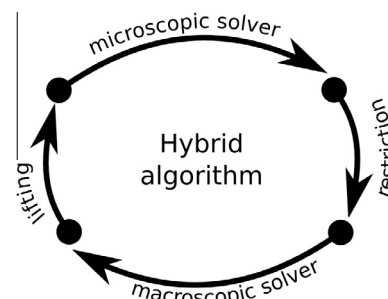


Fig. 1. Generic scheme of the continuum-atomistic hybrid algorithm.

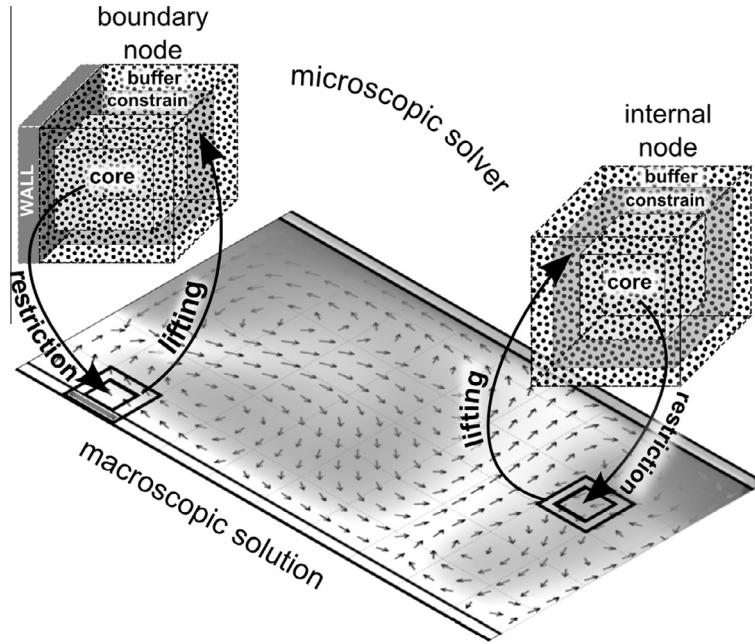


Fig. 2. Schematic representation of the macro-to-micro (lifting) and micro-to-macro (restriction) exchange of information for a boundary node and an internal node.

- Step 2. Lifting¹: knowing the macroscopic solution at the nodes, we can determine N sets of boundary condition to be applied to the microscopic solver.
- Step 3. Microscopic solvers: N molecular dynamics simulations are run.
- Step 4. Restriction: the molecular results are processed in order to extract the information required to update the macroscopic solver and start a new cycle.

Steps 1 and 3 are respectively the ‘pure’ continuum and molecular models; good general references, in this case, are respectively Landau and Lifshitz [22] and Allen and Tildesley [8]. Steps 2 and 4, however, are specific to hybrid models; in practice, each of these can be seen as an interface which regulates the exchange of information between the two solvers. The type of information, and how it is extracted from one model and introduced into the other, is precisely what characterizes a specific hybrid approach. In Sections 3 and 4 we discuss Steps 2 and 4 of our approach.

In the following sections we generally use lowercase letters (\mathbf{u} , \mathbf{r} , z , etc.) for the fluid variables related to the micro subdomain and uppercase letters to indicate the corresponding macroscopic variables (\mathbf{U} , \mathbf{R} , Z , etc.). The only exception is the temperature T ; since the lowercase t is used for time, the temperature for the micro-simulations is indicated with the Greek letter ϑ . For consistency with the units used in our MD code, all results are presented in reduced/normalized units as described in Appendix A.

3. The framed computational cell

Consistency between the microscopic and the macroscopic domains imposes certain conditions (e.g. streaming velocity, density and temperature) at the boundaries of the MD cell. If we think of the MD cell as a small window embedded into the macroscopic domain as in Fig. 2, we understand that, in the region where the two domains are ‘glued’ together, the outcome of the two models should match. Let us assume, for instance, that, as in the HMM,

the first-principle data the continuum model requires is the momentum flux calculated from the MD simulation and then passed to the macroscopic solver. The value of the momentum flux depends on the velocity gradient; this information is contained in the continuum domain, which, therefore, must be passed to the microscopic solver as a boundary condition. In order to achieve this goal, we use the ‘framed’ cell proposed by Hadjiconstantinou and Patera [16] (see Fig. 2). Both the MD cells and the continuum domain are, of course, three-dimensional, but in order to make the picture simpler a two-dimensional macroscopic flow is illustrated in Fig. 2. The internal region of the cell – the ‘core’ – is the region where the MD information is extracted and transmitted to the macroscopic model. The grey area is the ‘frame’ or ‘constrained region’; the region where the two models are coupled together and where the information from the macroscopic model is imposed on the microscopic model. The external region surrounding the frame is the ‘buffer’, which does not play any role in the exchange of information between the solvers but it allows the solution to relax between the external boundaries of the box and the frame.

Various ways of constraining the velocity in the MD simulation have been proposed. There is no unanimous opinion on what is the best option (see [27] for a review). Here, we use a simple velocity-shifting/velocity-rescaling technique. The frame is divided into bins and, at each time step, the molecular velocity distribution in the bin is rescaled and shifted so that the streaming velocity and temperature of the molecules contained in the bins match those of the continuum domain. The streaming velocity is defined as

$$\mathbf{u} = \frac{\sum_i m_i \mathbf{v}_i}{\sum_i m_i}, \quad (5)$$

and the temperature

$$\vartheta = \frac{\sum_i m_i (\mathbf{u} - \mathbf{v}_i)^2}{k_B N_v}, \quad (6)$$

where k_B is the Boltzmann constant and N_v the number of degrees of freedom of the system.

In Hadjiconstantinou and Patera [16], the constraint was achieved by using a ‘Maxwell Demon’ approach. In our case, however, we find that the velocity-shifting/velocity-rescaling strategy

¹ We borrow the terms ‘lifting’ and ‘restriction’ from the ‘equation free’ literature [20].

provides a smoother transition, especially for temperature profiles, between the frame and the core region. More details on this type of cell can be found in Drikakis and Asproulis [11].

The role of the buffer in the MD sub-domain is also important. There are, in fact, two types of conceptually different boundary conditions in the cell and they should not be confused.² The first conditions are the velocity and temperature profiles coming from the continuum domain. These are fundamental to the physics of the problem and are imposed in the frame. The second conditions are the external boundaries of the computational box. Since any molecular dynamic simulation is carried out in a computational box, a set of rules must be established to determine what happens if a molecule crosses the boundaries of the box. In the present case, these boundary conditions are only needed to contain all the molecules within the computational box and the molecules in the buffer are not considered in the calculation of any property transferred to the macroscopic solver.

We can restrain the molecules in the box by using periodic boundary conditions or artificial walls (force fields). Generally, we use periodic boundary conditions because, in this way, the profiles are smoother and, therefore, the thickness of the buffer region can be smaller. However, this solution is only possible in the internal nodes of the macroscopic domain. In the boundary nodes,³ where at least one of the cell faces is a boundary (e.g. solid wall, interface between two fluids, etc.), periodic boundary conditions cannot be used in the direction normal to the boundary. In this case, we need to use a force field or an artificial wall to confine the molecules. Here we use a reflecting wall, which bounces back the molecules in a mirror-like fashion when they cross the boundary. Fig. 3 shows the temperature (ϑ), the streaming velocity (u_x) and the density⁴ (ρ') profiles in an internal and in a boundary node reported in reduced units.

In Fig. 3, we show a version of the ‘framed’ cell that is modified for channel flows. The streaming velocity simplifies to u_x and the other components are zero. Since, in this case, the flow in the x and y direction is symmetric, the framed cell can be substituted by the ‘layered’ cell illustrated on the right side of Fig. 3. If the node is on the boundary, on one side we have the wall molecules but no frame and no buffer; on the other side, we have the constrained and the buffer ‘slices’, as before.

In Fig. 3(a) (for bulk node), the sudden jumps in the measured values between the constrained and buffer region bins are due to the fact that the cell is periodic and, consequently, the two buffer ‘slices’ are contiguous. Fig. 3(b) (for a wall boundary node) shows that on the non-periodic boundaries the density oscillates. This is due to molecular stratification and it is the correct behaviour in the proximity of a wall. In our boundary nodes, however, only one side has a real wall; the other side has an artificial wall that is only needed to confine the molecules. On the artificial wall, therefore, the oscillations are not physical and must be excluded from the core region. Sometimes, however, it is not easy to establish beforehand how deep into the cell the effects of these oscillations propagate. For this reason, the buffer of the boundary nodes should normally be larger. The method we propose (see Section 4) handles the boundary and the internal nodes differently. At the boundary nodes, all the necessary coupling information is not extracted from the core, but from bins close to the wall.⁵ Therefore,

² Confusion should be avoided also between the microscopic and macroscopic boundary conditions. The distinction discussed here concerns the microscopic boundary conditions.

³ With boundary nodes, we here refer to the nodes located at the boundaries of the macroscopic domain.

⁴ The density ρ' in Fig. 3 is the ratio ρ/ρ_0 , where ρ is the local density and ρ_0 the average density (here $\rho_0 = 0.8$).

⁵ Usually only the bin closest to the wall is considered, but, as discussed in Section 6, there are cases where also the second bin from the wall is necessary.

the only thing we have to pay attention to is that the perturbations generated by the two walls (the real and the artificial) do not overlap. With this method, there is no need to use larger cells near the wall. On the contrary, in this region, we can even use smaller cells.

4. Coupling with the macroscopic solver

After the MD simulations are completed, we need to extract information from the core of the cell and pass it to the macroscopic model.

In Ren and E [31], the momentum flux was calculated by means of the IK equation and then introduced into Eq. (1). Our method, instead, is based on the value of the Laplacian of the streaming velocity \mathbf{U} and, since we consider non-isothermal flows, the temperature T . We start from the balances of momentum and internal energy at steady-state [12]

$$\nabla \cdot \rho \mathbf{U} \mathbf{U} + \nabla p + \nabla \cdot \boldsymbol{\tau} - \rho \mathbf{g} = 0, \quad (7)$$

$$\nabla \cdot \rho \hat{U}_i \mathbf{U} + \nabla \cdot \mathbf{q} + p(\nabla \cdot \mathbf{U}) + \boldsymbol{\tau} : \nabla \mathbf{U} = 0, \quad (8)$$

where ρ is the fluid density, \mathbf{U} the macroscopic velocity, p the pressure, $\boldsymbol{\tau}$ the stress tensor, $\rho \mathbf{g}$ the external body forces acting on the fluid, \hat{U}_i the internal energy per unit volume and \mathbf{q} the heat flux.

We can express the shear stress tensor $\boldsymbol{\tau}$ in Eq. (7) and the heat flux vector in Eq. (8) respectively as

$$\boldsymbol{\tau} = -\mu \nabla \mathbf{U} + \boldsymbol{\Phi}, \quad (9)$$

and

$$\mathbf{q} = -\kappa \nabla T + \boldsymbol{\Psi}. \quad (10)$$

In Eq. (9), $-\mu \nabla \mathbf{U}$ is the ‘ideal’ or Newtonian part of the stress, μ the viscosity, and $\boldsymbol{\Phi}$ the deviation of the real stress from the ideal stress. In Eq. (10), T is the temperature, $-\kappa \nabla T$ is the ‘ideal’ or Fourier part of the heat flux, κ the thermal conductivity, and $\boldsymbol{\Psi}$ the deviation from the ideal flux. By introducing Eqs. (9) and (10) in, respectively, Eqs. (7) and (8), we obtain

$$\mu \nabla^2 \mathbf{U} = \nabla \cdot \rho \mathbf{U} \mathbf{U} + \nabla p - \nabla \cdot \boldsymbol{\Phi} - \rho \mathbf{g}, \quad (11)$$

and

$$\kappa \nabla^2 T = \nabla \cdot \rho \hat{U}_i \mathbf{U} - \nabla \cdot \boldsymbol{\Psi} + p(\nabla \cdot \mathbf{U}) + \boldsymbol{\tau} : \nabla \mathbf{U} = 0. \quad (12)$$

We collect all the terms on the right hand side of Eqs. (11) and (12) in two new functions

$$\Lambda(\mathbf{R}) = \frac{-\nabla \cdot (\rho \mathbf{U} \mathbf{U}) - \nabla p + \nabla \cdot \boldsymbol{\Phi} + \rho \mathbf{g}}{\mu}, \quad (13)$$

and

$$\Xi(\mathbf{R}) = \frac{\nabla \cdot \rho \hat{U}_i \mathbf{U} - \nabla \cdot \boldsymbol{\Psi} + p(\nabla \cdot \mathbf{U}) + \boldsymbol{\tau} : \nabla \mathbf{U}}{\kappa}, \quad (14)$$

where \mathbf{R} is the macroscopic position vector. We therefore have

$$\nabla^2 \mathbf{U} = \Lambda(\mathbf{R}), \quad (15)$$

$$\nabla^2 T = \Xi(\mathbf{R}). \quad (16)$$

At this point, we only need a way to estimate $\Lambda(\mathbf{R})$ and $\Xi(\mathbf{R})$ from molecular dynamics. The simplest choice is to approximate these functions with the average microscopic Laplacian:

$$\langle \nabla^2 \mathbf{u} \rangle_{\mathbf{r}=\mathbf{R}} \cong \Lambda(\mathbf{R}) \quad (17)$$

$$\langle \nabla^2 \vartheta \rangle_{\mathbf{r}=\mathbf{R}} \cong \Xi(\mathbf{R}), \quad (18)$$

and, therefore,

$$\nabla^2 \mathbf{U}(\mathbf{R}) \cong \langle \nabla^2 \mathbf{u} \rangle_{\mathbf{r}=\mathbf{R}}, \quad (19)$$

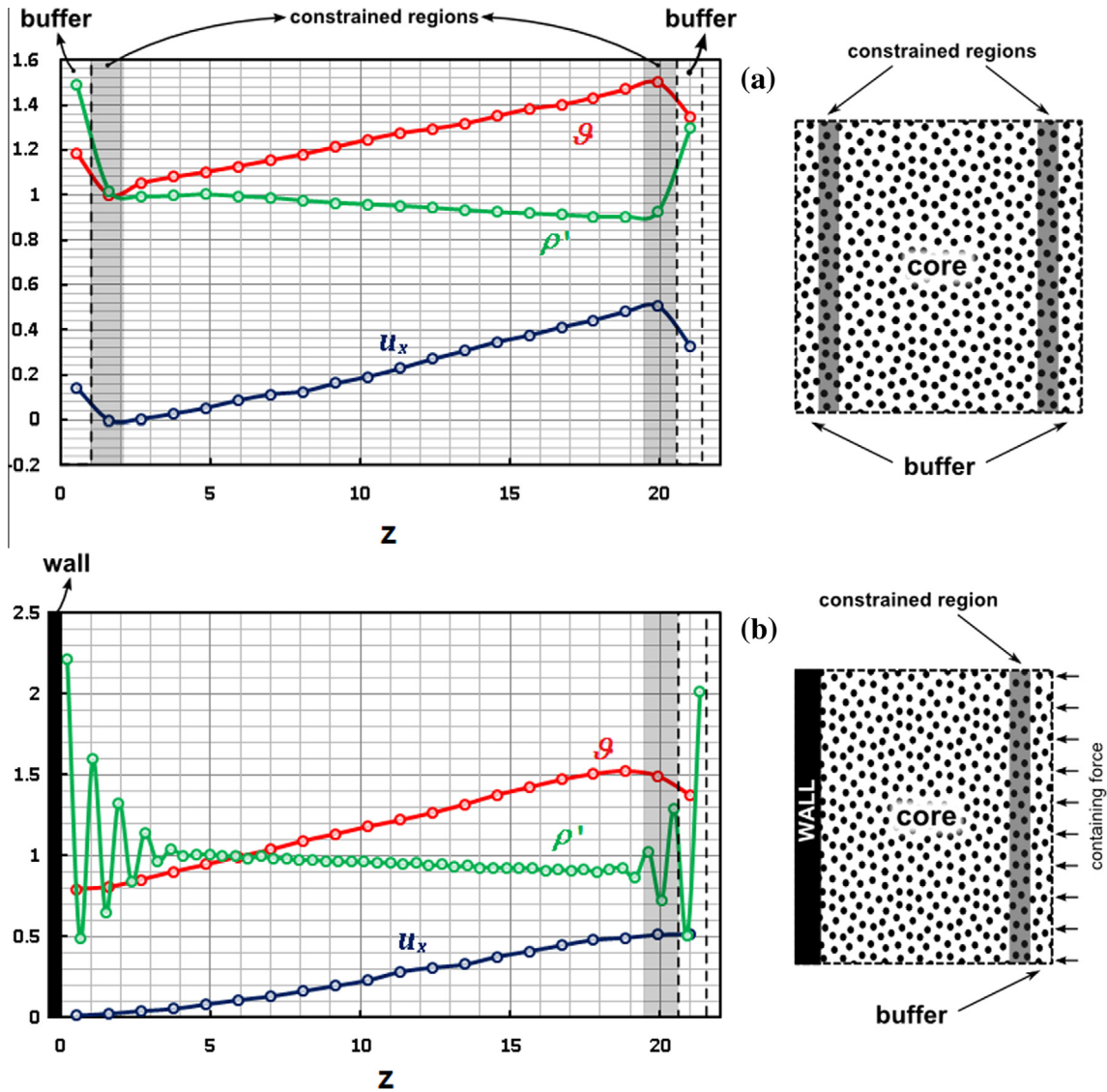


Fig. 3. Fluid properties profiles in two 'layered' cells; the first (a) is for a bulk node, the second (b) is for a boundary node. Simulation are carried out in a $21.5 \times 5.4 \times 21.5$ reduced units box ($20 \times 5 \times 20$ unit cells) with 2000 Lennard–Jones molecules and overall density $\rho = 0.8$. Averaged profiles are calculated from 40,000 time steps ($\Delta t = 0.05$) after 2000 equilibration time steps.

$$\nabla^2 T(\mathbf{R}) \cong \langle \nabla^2 \vartheta \rangle_{\Gamma=\mathbf{R}} \tag{20}$$

According to Eqs. (19) and (20), the Laplacians of the macroscopic velocity \mathbf{U} and temperature T at the node located at \mathbf{R} can be determined from the averaged Laplacians of the microscopic streaming velocity \mathbf{u} and temperature ϑ obtained from molecular dynamics.

Eqs. (19) and (20) are valid everywhere except at the boundary nodes, where the macroscopic boundary conditions at the walls should be introduced. Since the traditional no-slip velocity and no-jump temperature boundary condition does not generally apply to micro and nanoflows [19], we must extract this information from the MD simulations too. In this case, we calculate directly the value of the velocity and temperature, instead of their Laplacians, in the bins adjacent to the walls. Thus, at the nodes located at the walls, Eq. (19) is replaced by

$$\mathbf{U}(\partial\mathbf{R}) \cong \langle \mathbf{u} \rangle_{\Gamma=\partial\mathbf{R}}, \tag{21}$$

and Eq. (20) by

$$T(\partial\mathbf{R}) \cong \langle \vartheta \rangle_{\Gamma=\partial\mathbf{R}}, \tag{22}$$

where $\partial\mathbf{R}$ indicates the boundary nodes.

5. Validation case: non-isothermal channel flow

It is common practice to validate hybrid methods against benchmark fluid dynamics problems, such as Poiseuille, Couette or lid driven cavity flows (see [27] for a review). The equivalent full-MD solutions of these problems, in fact, can be easily calculated and compared with the hybrid results in order to assess the accuracy of the method proposed.

Since our approach can be applied to both isothermal and non-isothermal flows, we need to take into account a validation case more complex than usual in order to test the hybrid model in situations where momentum and heat transfer are coupled together. As benchmark, therefore, we consider an incompressible fluid in a micro or nano-channel put in motion by a gravity-like body force \mathbf{g} in the x direction, while the walls are kept at different velocities and temperatures (Fig. 4). Besides our validation purposes, this represents a good choice also for practical reasons. There are, in fact, many micro and nano-channel systems that would benefit from

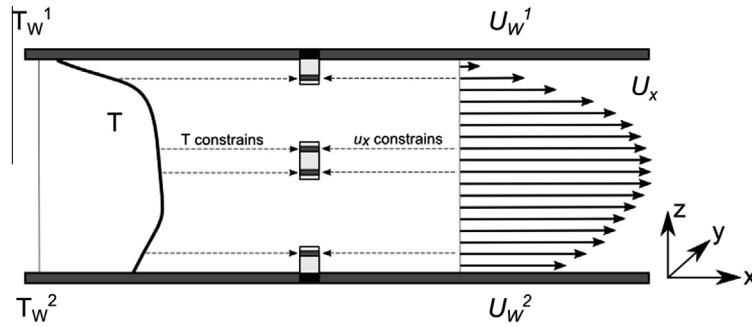


Fig. 4. Channel flow with bounding walls at two different velocities and temperatures. Two wall nodes and one internal node ('layered' MD cells) are illustrated.

the extension of the HMM to non-isothermal flows. Among them, computer chip cooling deserves special attention. With chip power densities increasing beyond the air cooling limit, liquid cooling methods based on water (e.g. [7] or metals with low melting point (e.g. [23] are currently being investigated. The accurate calculation of liquid refrigerant flows in non-isothermal micro-channels is important for the correct assessment of the shear-stress/temperature and heat-flux/temperature relation, and the temperature jump across the walls. The validation case presented in this section represents a good model for this type of practical applications.

In the case considered here, the flow is at steady-state. For simplicity, we assume periodicity in the y direction and we neglect the boundary effects at the entrance and at the exit of the channel. Under these assumptions Eqs. (19) and (20) simplify respectively to

$$\left\langle \frac{\partial^2 U_x}{\partial Z^2} \right\rangle_Z = \left\langle \frac{\partial^2 u_x}{\partial z^2} \right\rangle_{z=Z}, \quad (23)$$

and

$$\left\langle \frac{\partial^2 T}{\partial Z^2} \right\rangle_Z = \left\langle \frac{\partial^2 \vartheta}{\partial z^2} \right\rangle_{z=Z}. \quad (24)$$

At the boundary nodes, we have

$$(U_x)_{\partial Z} = \langle u_x \rangle_{z=\partial Z}, \quad (25)$$

and

$$(T)_{\partial Z} = \langle \vartheta \rangle_{z=\partial Z}. \quad (26)$$

The macroscopic velocities and temperatures are obtained by solving numerically Eqs. (23)–(26). Here, we use a simple finite difference scheme:

$$\left(\frac{\partial^2 Y}{\partial Z^2} \right)_{Z_i} \approx \frac{Y_{i-1} - 2Y_i + Y_{i+1}}{h^2}; \quad i = 1, \dots, N-2, \quad (27)$$

where Y can be either U_x or T from Eqs. (23) and (24), h is the distance between two nodes and the values of the boundary nodes $i = 0$ and $i = N - 1$ come respectively from Eqs. (25) and (26).

Before we discuss numerical results, we use Fig. 5 to summarize how, in our specific case, the hybrid algorithm works step by step (in Fig. 5, only a single boundary node and a single internal node are shown).

- We start with an estimate of the velocity $U_x(Z)$ and the temperature $T(Z)$, which is calculated at each node Z_i of the continuum domain (initialization).
- A spline interpolation⁶ is performed and the values of $U_x(Z_i - \Delta z_{MD})$, $U_x(Z_i + \Delta z_{MD})$, $T(Z_i - \Delta z_{MD})$ and $T(Z_i + \Delta z_{MD})$ are

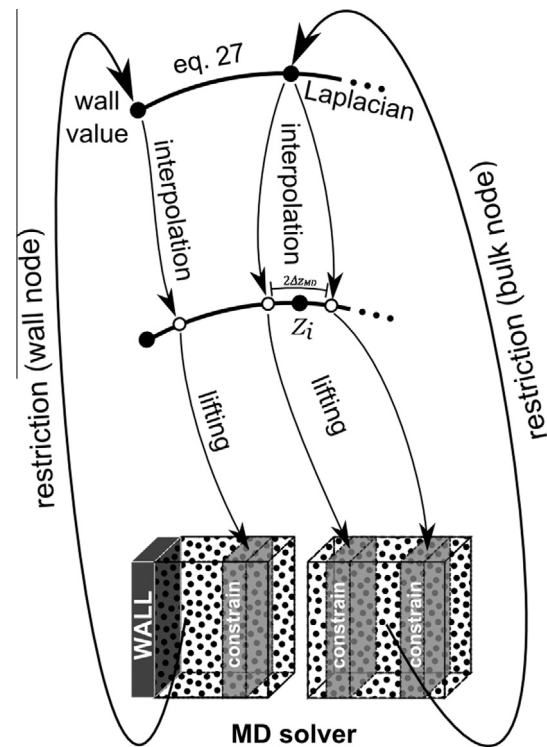


Fig. 5. The Laplacian hybrid cycle illustrated for a boundary node and a bulk node ('layered' cell).

- calculated in the neighbourhood of each node Z_i (interpolation).
- These values are used to constrain the velocity and the temperature in a 'layered' MD cell with thickness $2\Delta z_{MD}$ (lifting).
- The MD simulations are carried out in the neighbourhood of every node (MD solver).
- For each internal node, the Laplacian (or, in this particular case, the curvature) and, for the boundary nodes, the wall-value in the bin closest to the wall of both velocity and temperature are extracted from the molecular results. In order to do this, we approximate (using least squares minimization) the $\vartheta(z)$ and $u_x(z)$ profiles to a parabola⁷ and calculate the second derivative. These values are then introduced in Eq. (27) (restriction).
- Eq. (27) is solved numerically and new macroscopic values are calculated at each node (macroscopic solver).

⁶ For the velocity, natural cubic splines are sufficient. For the temperature, we found that Akima splines [9] improve the accuracy of the algorithm, especially when the T profile is relatively flat in the centre of the channel.

⁷ In case of two-dimensional or three-dimensional flows, bi-quadratic or tri-quadratic interpolation should be used.

The procedure from (b) to (f) is then repeated and, at each coupling step k , new updated macroscopic profiles $U_x^k(Z)$ and $T^k(Z)$ are determined. Normally, this type of procedure continues until a convergence criterion of the type $|Y^{k-1}(Z) - Y^k(Z)| < \epsilon$ is met. Because of the intrinsically noisy nature of the MD simulation, however, the profiles Y_x^k oscillate. It is more effective, therefore, to run the solution for a certain number of cycles and calculate the average. The noise fluctuations can be reduced by relaxing the solution e.g.

$$Y^k = (1 - \alpha)Y^{*k} + \alpha Y^{k-1}, \tag{28}$$

where Y^{*k} is the velocity or temperature calculated by Eq. (27) at step k , Y^k is the final value after relaxation and α is the relaxation factor. In the results discussed in the next section, we used a relaxation factor of 0.5 for both velocity and temperature.

6. Results

6.1. General case

We consider a channel with a thickness of 80 reduced units and stochastic walls [33,30]. Temperature and velocity of the first wall

are respectively $U_w^1 = 0$ and $T_w^1 = 1$ in reduced units, while, on the second wall, they are $U_w^2 = 3$ and $T_w^2 = 1.5$. An additional body-force with acceleration $g_x = 0.01$ is added to the molecules. Stochastic walls are not always the most realistic, but, since we are here equally employing them for both the validating (full-MD) and validated (hybrid) system, the question of their generic reliability is irrelevant.

In the hybrid calculation, the continuum solution is calculated at $N = 6$ nodes. The size of each MD box is $5 \times 5 \times 10$ unit cells ($5.4 \times 5.4 \times 10.8$ reduced units) with 250 Lennard–Jones (LJ) atoms ($\rho = 0.8$). Since reduced units are used, the value of the LJ parameters is unitary. Each MD simulation was run for 6000 time steps ($\Delta t = 0.005$), where the first 1000 are for equilibration. The $\vartheta(z)$ and $u_x(z)$ curvatures are calculated every 1000 time steps. At the end of the simulation, therefore, we have five values of the curvatures, which are averaged before being introduced into Eq. (27). The atomistic domain is divided into 10 longitudinal bins in the z -direction. The two lateral bins are the buffer ‘slices’, while the two contiguous to the buffer are the frames where velocity and temperature are constrained.

The hybrid results are compared with full MD results calculated for the same channel. The MD domain is a box of $10 \times 10 \times 75$ unit

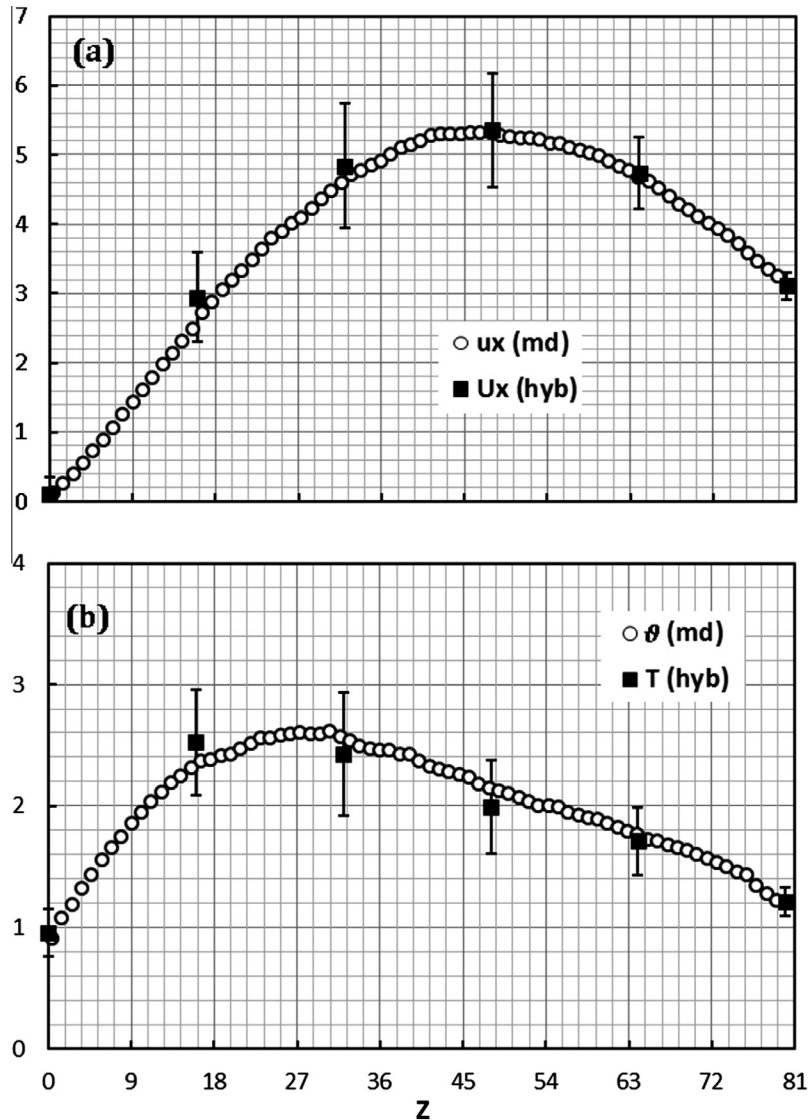


Fig. 6. Comparison between velocity (a) and temperature (b) profiles calculated using molecular dynamics (white circles) and the hybrid algorithm proposed in this study (black squares).

cells ($10.8 \times 10.8 \times 81$ reduced units) with a total of 7500 atoms. The simulation is run for 200,000 simulation steps and average profiles stored every 10,000 steps. Temperature is not controlled in order to allow viscous dissipation to affect the temperature profile.

Fig. 6 shows the comparisons between MD and hybrid results for velocity (Fig. 6(a)) and temperature (Fig. 6(b)). The agreement between the hybrid results and the pure MD simulation is very good both for temperature and streaming velocity. In both cases, the error bars of the hybrid data are relatively high, but acceptable. This depends on the fact that the MD part of the algorithm introduces fluctuations in the results. Numerical noise exists also in the pure MD simulation, but the total number of atoms is higher and, therefore, fluctuations are smaller. Error bars for the full MD data are not reported, but they are considerably smaller than the hybrid ones. More details on the fluctuations (and how to reduce them) are given below.

Other issues that deserve attention are the velocity and temperature discontinuities at the boundaries. Stochastic walls do not generate slip as Fig. 6(a) shows. They do, however, produce a temperature jump. In this specific case, for instance, we imposed $T_w^1 = 1$ and $T_w^2 = 1.5$ at the walls but, as Fig. 6(b) indicates, the temperature in the bin closest to each wall is slightly lower (respectively 0.9 and 1.2). In Fig. 6, the effect of the temperature jumps is rapidly dominated by the presence of viscous dissipation but, without this dissipation, they would involve a larger portion of the fluid, as discussed in the next section.

6.2. Wall correction

In the previous example, the effect of the temperature jump at the walls is limited to the first bin near the wall. There are situations, however, where more bins are necessary. In Fig. 7, for instance, the temperature profile between two walls at different temperatures ($T_w^1 = 1$ and $T_w^2 = 2$) is reported. All the other simulation conditions (for both hybrid and full-MD cases) are the same as in Fig. 6, but here there are no body-forces ($\mathbf{g} = 0$) and neither wall moves ($U_w^1 = U_w^2 = 0$); the fluid, therefore, is at rest and heat is only transferred by conduction. Basically, we have here the classic one-dimensional heat equation at steady state, whose analytical (macroscopic) solution is simply a straight line connecting the wall temperatures. With the exception of the wall regions, the full MD solution confirms this linear behaviour (see Fig. 7). In this case, however, the temperature jumps at the walls are not strictly limited to the solid–fluid interface as in Fig. 6, but involve a region of approximately two molecular lengths. The imposed temperature at the walls is $T_w^1 = 1$ and $T_w^2 = 2$; in the first bin next to the walls, it drops to respectively 0.8 and 1.6. But, unlike the previous case, the temperature keeps falling and in the second closest bin it goes approximately to 0.7 and 1.3 respectively. When we calculate the hybrid solution, if we simply use the method described in Section 4 (in which only the first bin closest to the walls is considered), we obtain the first profile (triangular markers) in Fig. 7. The calculation of only one value of the temperature in the bin nearest to the wall is therefore not enough and does not guarantee, in this

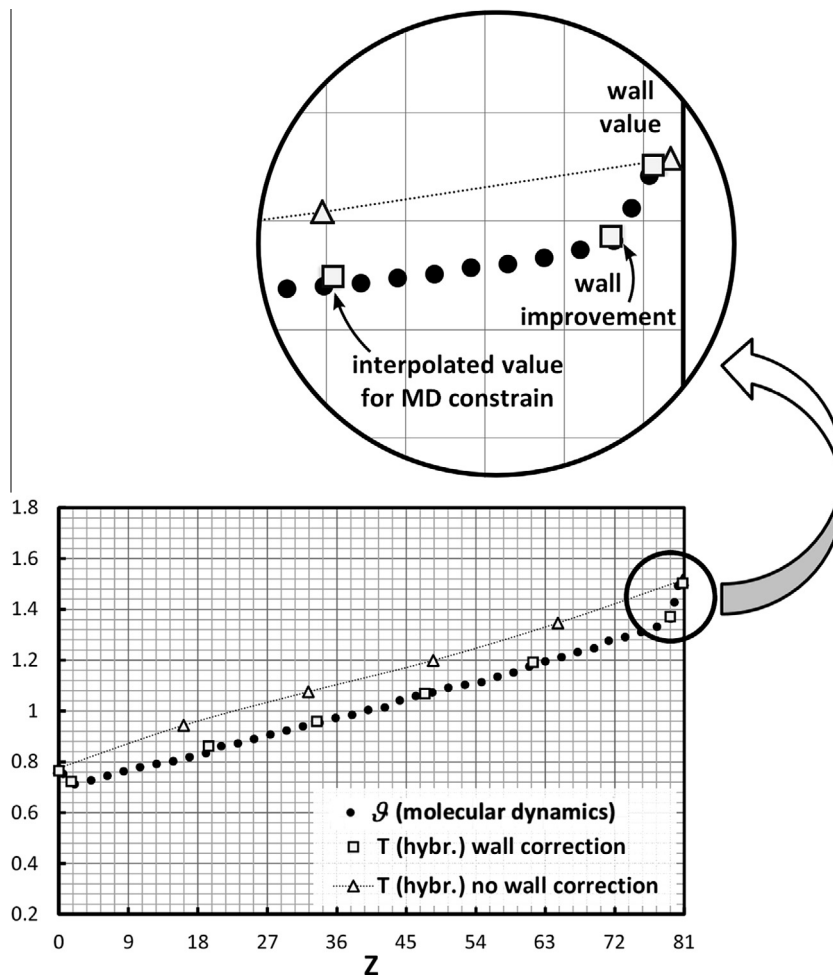


Fig. 7. Comparison between full MD temperature profiles and hybrid results calculated with and without wall correction in the case of heat transfer with fluid at rest.

case, a correct temperature profile. To correct this problem, we also consider a second temperature point at the wall (the second nearest bin referred as ‘wall improvement’ in Fig. 7). By using this simple method, the hybrid solution improves considerably, as Fig. 7 (square markers) indicates. In Fig. 7, we only report the average results. Since the two hybrid profiles are close to each other, the error bars would make Fig. 7 difficult to read. The fluctuations, however, are of the same relative magnitude as in Fig. 6.

6.3. Viscous dissipation

The velocity profile in Fig. 6(a) is not parabolic because part of the energy necessary to maintain the velocity gradient degrades to heat through viscous dissipation. As a consequence, the temperature and, therefore, the viscosity are not constant. The same phenomenon can be observed, at the nanoscale, also in the case of Couette flow. Fig. 8 reports the case where no gravity-like force is considered; the two walls have different velocities ($U_w^1 = 0$ and $U_w^2 = 4$) but the same temperature ($T_w^1 = T_w^2 = 1$). The remaining simulation conditions (for both hybrid and full-MD cases) are the same as in Fig. 6 with the difference that, here, the MD simulations of the hybrid model are run for 60,000 time steps instead of 6000. This does not mean that the calculation of the Couette flow requires a higher accuracy than the cases previously

investigated. We simply want to show, as discussed below, that increasing the statistical accuracy of the single MD steps reduces the overall fluctuations of the hybrid method.

The results are illustrated in Fig. 8. The analytical (macroscopic) solution of the Couette flow problem would be a straight line connecting the two wall velocities. Since in the case of stochastic walls the wall-slip is small, we do not have, for the velocity, the issue of the wall correction as for the temperature (see Section 6.2). The full-MD results, however, do not show a perfectly linear behaviour (Fig. 8(a)). This is due to the viscous dissipation, which heats the fluid and raises the temperature (Fig. 8(b)). In a macroscopic channel, the amount of heat generated in this way would be negligible, but in a nano-channel the situation is different. Since the fluid is not isothermal, the viscosity in the channel is not constant so the velocity profile is not perfectly linear. This is particularly true near the walls where the temperature gradients are higher. The hybrid results again match very well the full-MD data, confirming the effectiveness of our method.

In Fig. 8, we report, as in Fig. 6, the temperature and velocity values at the nodes (black markers). In this case, however, we also show the values in the constrained region (grey markers) of the ‘layered’ cell. This gives an idea of the relative size of the MD cells used in the hybrid model. In Fig. 8, the wall improvement (see Section 6.2) is used only for the temperature profile (the two darker

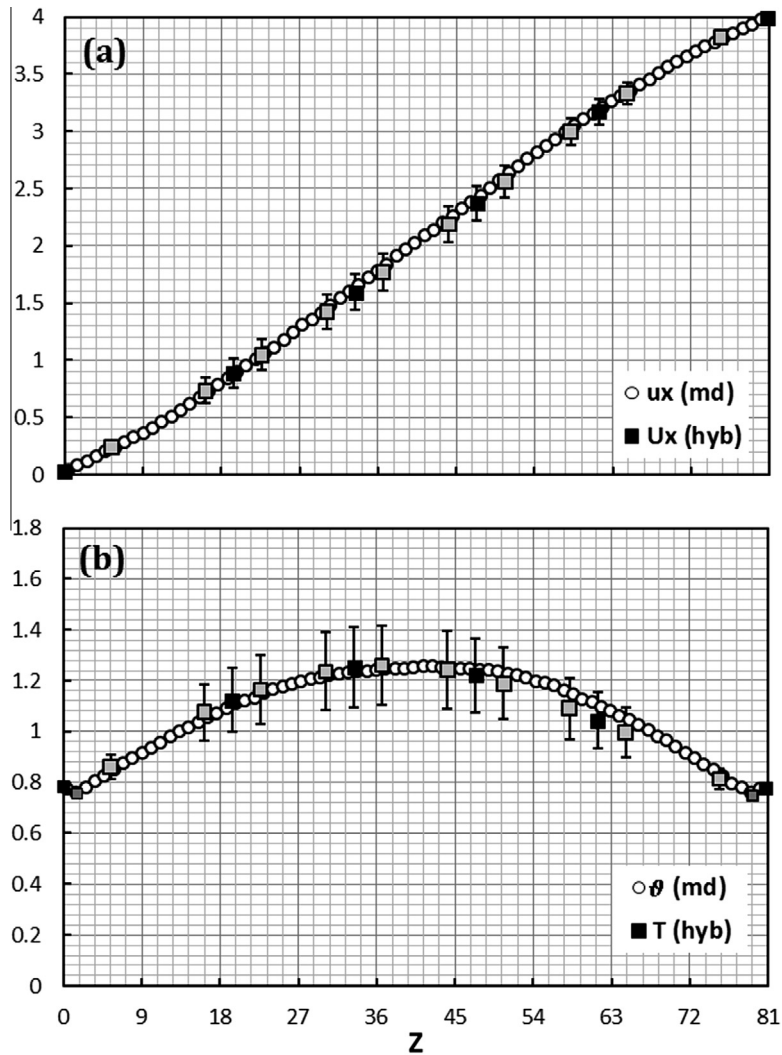


Fig. 8. Velocity (a) and temperature (b) comparisons between full MD and hybrid results for Couette flow with viscous dissipation.

grey and slightly smaller markers near the walls in Fig. 8(b)). For the velocity (Fig. 8(b)), no wall improvement is necessary because, as already mentioned, stochastic walls do not produce slip for liquid flows.

The error bars in Fig. 8 are considerably smaller than those in Fig. 6. This is due to the fact that, in this case, the MD simulations of the hybrid algorithm are run for 60,000 time steps instead of 6000. This implies higher accuracy of the microscopic solver and, consequently, smaller overall fluctuations in the hybrid solution. The price to pay for this improvement is, of course, a greater com-

putational time: a trade-off between accuracy and computational time is, as usual, required.

6.4. Isothermal case

Up to now, we have considered non-isothermal cases, but we can also test the accuracy of our method for isothermal flows, which require only the momentum equation to be solved. Fig. 9 shows the MD results for a $5 \times 5 \times 75$ unit cells box ($5.4 \times 5.4 \times 81$ reduced units), with density $\rho = 0.8$ and

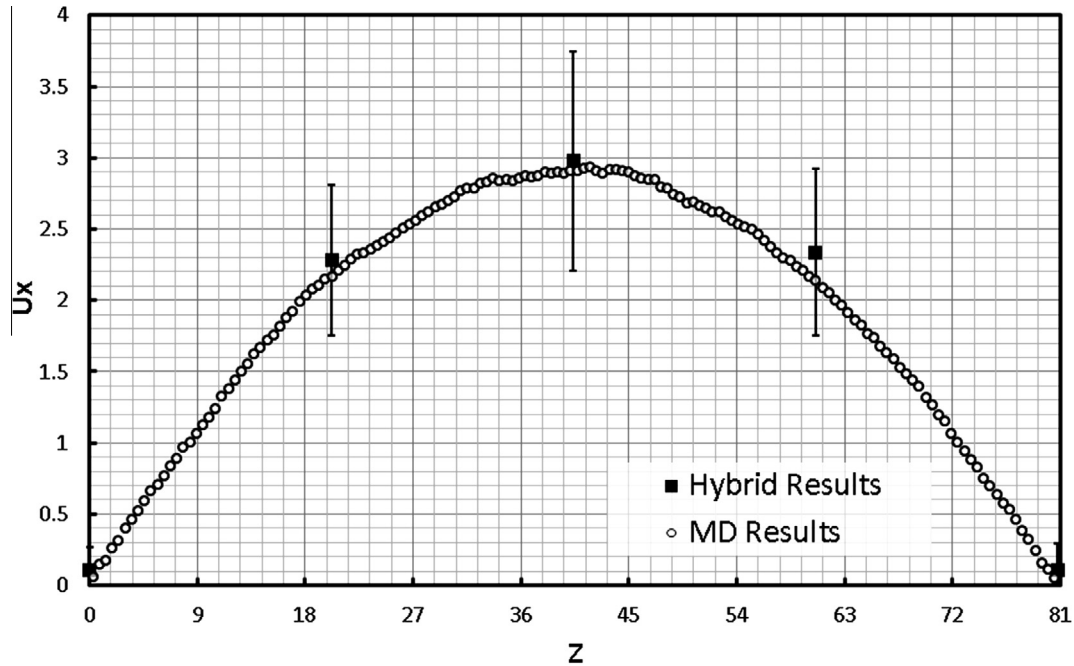


Fig. 9. Comparisons between full MD and hybrid velocity profiles for isothermal Poiseuille flow.

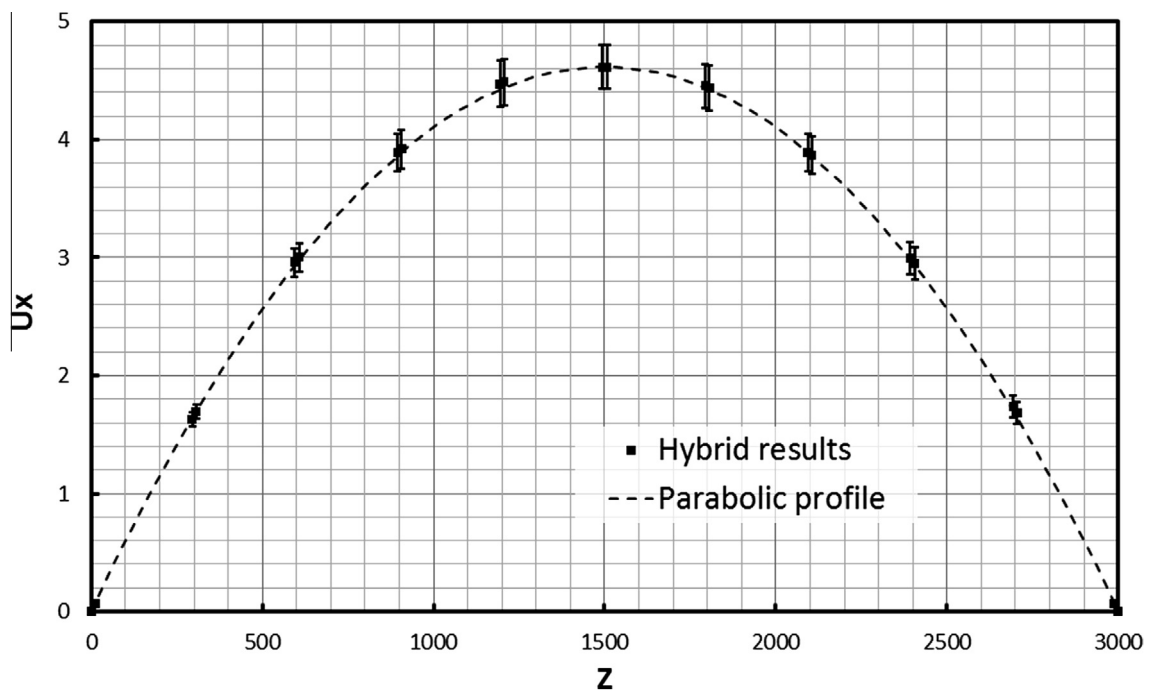


Fig. 10. Comparison between the hybrid results and the parabolic profile for a large isothermal channel.

$g_x = 0.0075$; the velocity of both (stochastic) walls is $U_w^{1,2} = 0$ and the temperature is $T_w^{1,2} = 1$. The domain in the z -direction is divided into 150 bins. We run two simulations for 250,000 time steps ($\Delta t = 0.005$). In the first one, no temperature control is implemented and the temperature rises slightly ($T \approx 1.5$ in the centre of the channel). In the second one the dissipation heat is removed with a velocity rescaling ($T = 1$) every 10,000 time steps so that the average temperature remains approximately constant in the channel. The streaming velocity is not particularly affected by the temperature control. Fig. 9 shows the velocity profile for the temperature-controlled case, but the profile for the temperature-non-controlled case is very similar. Concerning the hybrid results, we run the simulation under the same conditions of Fig. 6, but $g_x = 0.0075$, $U_w = 0$ on both walls, $L_z = 81$ and $N = 5$. The fluid temperature in the constrained regions of every node is fixed to $T = 1$. Once the temperature is controlled in the ‘frames’, and since the MD boxes used in the hybrid scheme are small, the temperature does not rise particularly and no further temperature control is necessary. While for Figs. 6–8 a relaxation factor (see Eq. (28)) $\alpha = 0.5$ was used, here no relaxation ($\alpha = 0$) is employed.

Fig. 9 shows the comparison between our hybrid results and the parabolic profile. As already discussed, the hybrid results are rather noisy and this can be seen from the error bars. While for Fig. 8 we used more MD time steps than for Fig. 6 to show how this decreases the fluctuations, here we did not use a relaxation factor

in order to show how this increases the fluctuations. Despite the higher noise, however, the algorithm remains stable and the averages in Fig. 9 closely follow the MD profile.

6.5. Macroscopic validation

So far, we validate our method by comparing hybrid and full-MD results. This is, as already mentioned, common practice and, up to now, we conformed to it. In general, however, this type of validation is restrictive because, in this way, we only test the model for small systems, despite the fact that the reason why we study hybrid algorithms in the first place is to be able to simulate systems much larger than MD is capable of simulating within reasonable computational times. Moreover, in order to show a visually more effective comparison with the MD results, we used more nodes than necessary in our previous hybrid simulations. Therefore, the computational speed-up of the hybrid algorithm with respect to MD cannot be always fully appreciated from these particular examples.

Normally, there is not much of a choice since we are restricted by the relatively limited size of the full-MD suitable cases. An isothermal Lennard–Jones fluid in a large channel, however, can be considered, with a certain approximation, Newtonian if the strain-rate does not vary excessively in the domain. We can, therefore, take advantage of this circumstance and compare our results

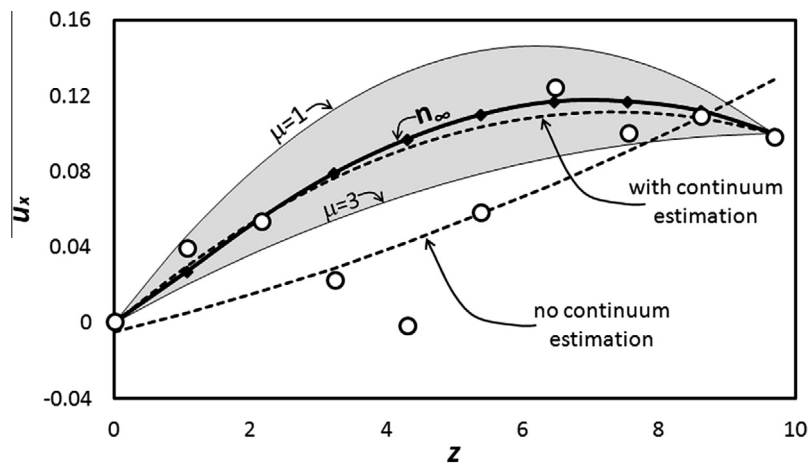


Fig. 11. Fitting improvement by using the continuum estimation.

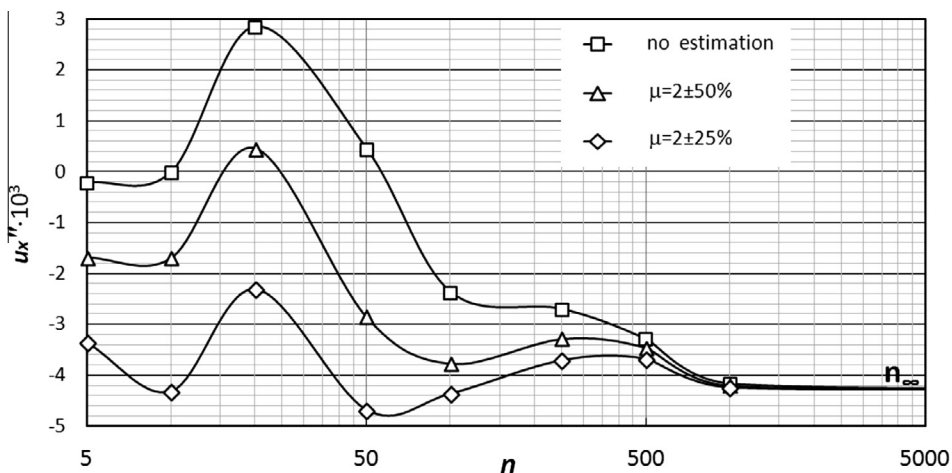


Fig. 12. Streaming velocity curvature versus number of samples.

with the corresponding parabolic profile in order to assess the accuracy of the hybrid model for larger channels. Here, we simulated an isothermal channel ($T = 1$, $N = 11$, $U_w^{1,2} = 0$, $g_x = 10^{-5}$, $\alpha = 0.1$ and $10 \times 10 \times 10$ unit cells per node; other conditions are the same as in Fig. 8) of size $L_z = 3000$ reduced units. If we consider that in many practical cases the σ parameter of the Lennard–Jones potential is between 3 and 6 Å, this means that we are now simulating a channel with a width between 1 and 2 μm . The hybrid calculation took only a few hours on a simple Desktop computer (Intel i-7-26000 QuadCore 3.4 GHz, 8 GB RAM), but it would be considerably harder to run a full MD simulation. We cannot even quantify how large is the speed-up gain in this case because for MD it would be impossible to simulate a channel of this size without the aid of a supercomputer.

Fig. 10 shows, as expected, the clear parabolic profile of the hybrid data. If, from the fitting parabola, we determine the viscosity, we find a value of 1.9 reduced units, which is perfectly consistent with the non-equilibrium equation of state for Lennard–Jones fluids [2]. The fluctuations are here kept under control by using longer MD steps, as in Fig. 8, and a lower relaxation value of 0.1. In Fig. 10 only the values at the constrained regions are reported. In this case, the MD cells have the same length of the previous examples, but, since here the overall size of the channel is much larger, they look considerably smaller. Moreover, we employed in this case a smaller number of nodes relative to the size of the macroscopic domain. In general, the larger the system the more efficient is the hybrid computation because the optimal ratio between N and the size of the macroscopic domain decreases.

We should not be misled by the apparent simplicity of this example. The parabolic profile is naturally recreated by the hybrid solver without any *a priori* assumption on the stress. The fact that from the intrinsic complexity of the intermolecular interactions we arrived, as expected in this particular situation, to the relatively simple Newton law of viscosity is a significant result and a good test for our method.

7. Conclusions

We have proposed a new algorithm within the HMM/PWC framework, which is based on the calculation of the velocity/temperature Laplacians at internal nodes and the velocity/temperature values at boundaries. The algorithm performed very well in all test cases considered. Since our hybrid method needs the estimation of second derivatives from MD profiles, a certain level of fluctuation in the result is expected. We discussed which factors affect the fluctuations and how they can be dramatically reduced by using higher relaxation factors and longer MD steps. A third way of reducing the fluctuations is discussed in Appendix B.

There are three main advantages in our method. Two have already been mentioned: the algorithm works for non-isothermal flows and it does not need the calculation of the Irving–Kirkwood fluxes. The third advantage is that direct knowledge of the source terms (\mathbf{g} in Eq. (7) and $\boldsymbol{\tau}:\nabla\mathbf{U}$ in Eq. (8)) is not required since their effect is automatically included in $\boldsymbol{\Lambda}$ and $\boldsymbol{\Xi}$ (see Eqs. (13) and (14)). At first sight, the importance of this fact can be underestimated. The value of the gravity-like term \mathbf{g} , for instance, should be perfectly known so its calculation does not seem to be an issue. In practical microfluidic applications, however, the flow is not usually generated by simple gravity-like forces as in Eq. (7). More often, the driving force comes from electro-osmotic, piezoelectric, or magnetic effects [1,17]. While it is possible to implement these in an MD simulation, their effect on the continuum model cannot be concentrated in a simple constant as \mathbf{g} in Eq. (7). This makes our method viable also when these forces are not perfectly known at the macroscopic level, and this is clearly an advantage in some practical situations. For the non-isothermal case, the benefit is

even more obvious since the $\boldsymbol{\tau}:\nabla\mathbf{U}$ term is more complex than \mathbf{g} . The calculation of the relatively complicated viscous dissipative term (enclosed in $\boldsymbol{\tau}:\nabla\mathbf{U}$) is, in our model, not required, but emerges automatically from the Laplacian.

There are, moreover, some features of the algorithm that make it attractive from a practical point of view. Both the discrete and the continuous parts require minimal changes from, respectively, standard MD codes and partial differential equations (PDE) numerical solvers. Concerning the atomistic model, for instance, since rigid periodic boxes are employed, the only real modification with respect to a standard MD code is the presence of the constrain region. Concerning the continuum solver, once the molecular source term is introduced into Eqs. (21) and (22), we simply obtain the Poisson equation, which is one of the most studied PDEs, and many numerical solvers are available for its solution.

The main purpose of this article is to introduce, for the first time, the Laplacian method and, for this reason, the issue of the algorithm numerical efficiency is not directly addressed here. There are at least three possible lines of improvement (namely, the assessment of the minimal number of nodes for a given accuracy, the introduction of a “seamless” strategy as proposed by E et al. [15], and a more robust estimation of the Laplacian), but they are left for future work. Despite this, Section 6.5 provides an idea of the huge computational savings that are expected in large geometries by using the Laplacian method instead of the full MD.

Acknowledgments

This research is financially supported by EPSRC Programme Grant EP/I011927/1.

Appendix A. Reduced units

In our simulations, we use the Lennard–Jones potentials

$$V_{ij}^{LJ} = 4\epsilon \left[\left(\frac{\sigma_{LJ}}{r_{ij}} \right)^{12} - \left(\frac{\sigma_{LJ}}{r_{ij}} \right)^6 \right], \quad (\text{A.1})$$

where ϵ is the characteristic energy level of the potential, σ_{LJ} the molecular length scale and r_{ij} the distance between atoms i and j .

In this case, the most appropriate system of units adopts σ_{LJ} , m (the mass of the molecule) and ϵ as units of length, mass and energy, respectively. All other variables are determined in relation to these e.g.

$$r^* = \frac{r}{\sigma_{LJ}} \quad (\text{length}), \quad (\text{A.2})$$

$$t^* = \frac{t}{\sqrt{\epsilon/m\sigma_{LJ}^2}} \quad (\text{time}), \quad (\text{A.3})$$

$$\mathbf{u}^* = \mathbf{u} \sqrt{\frac{m}{\epsilon}} \quad (\text{velocity}), \quad (\text{A.4})$$

$$T^* = T \frac{k_B}{\epsilon} \quad (\text{temperature}), \quad (\text{A.5})$$

$$f^* = f \frac{\sigma_{LJ}}{\epsilon} \quad (\text{force}), \quad (\text{A.6})$$

$$P^* = P \frac{\sigma_{LJ}^3}{\epsilon} \quad (\text{pressure}), \quad (\text{A.7})$$

$$\rho^* = \rho \frac{\sigma_{LJ}^3}{m} \quad (\text{density}), \quad (\text{A.8})$$

$$\mu^* = \mu \frac{\sigma_{ij}^3}{\sqrt{m\epsilon}} \quad (\text{viscosity}), \quad (\text{A.9})$$

$$g^* = g \frac{m\sigma_{ij}}{\epsilon} \quad (\text{acceleration}), \quad (\text{A.10})$$

where each dimensional unit has a corresponding (asterisked) reduced unit, which is dimensionless. In order to simplify the notation, the asterisk is not used in this article; nonetheless all the numerical values are always reported in reduced units.

Appendix B. Continuum estimation for noise reduction

The calculation of the Laplacian is central in our method and, at the same time, the main source of noise. In this article, we used two ways of reducing the fluctuations: longer MD steps and the relaxation factor. We saw that the price to pay for using these methods consists in longer computational times. There is, however, a third idea, which does not have this drawback, but, on the other hand, requires an initial approximation of the solution. In the validation cases discussed in this study (Section 6), we did not use this third approach because we wanted to test our method in the worst possible scenario, where no approximation is available. In general, however, this can be obtained by calculating the continuum solution. With “continuum solution” in this case, we do not mean the solution calculated during the macroscopic step of the hybrid algorithm, but the solution we would obtain by solving only the macroscopic equations without any microscopic refinement. We could, for instance, close Eqs. (7) and (8) with, respectively, Newton’s law of viscosity and Fourier’s law of heat conduction and look for a reasonable estimation of the viscosity and the thermal conductivity.

A practical example can help us explaining this method. We take into consideration a framed computational cell composed of $10 \times 10 \times 12$ unit cells. The streaming velocity and temperature are constrained in the 2nd and 11th bin in the z direction to, respectively, $U_1 = 0$ and $T_1 = 1$, and $U_2 = 0.1$ and $T_2 = 1$. The acceleration g_x is fixed to 0.01, while the remaining parameters are the same of Fig. 5. While the simulation progresses, we sample every 50 time steps the molecular velocities. When a certain number n of samples is accumulated (we choose $n = 5, 10, 20, 50, 100, 250, 500, 1000$ and $100,000$), we calculate the average streaming velocity in each bin. The white circles in Fig. 11, for instance, represents the velocity values at $n = 50$; they are significantly scattered because they are based on a small number of samples. As n increases, the velocity profile becomes more accurate. The profile labelled with “ n_∞ ” (black diamonds) represents the velocity calculated with a very high number of samples ($n = 100,000$). Since in this case a high number of samples are used, the data follow a well-defined curve as indicated in Fig. 11. If we increase n beyond this value, no noticeable effect on the velocities is observed and this is the reason why we named this profile n_∞ .

These velocity values (white circles in Fig. 11) are then fitted to a parabola whose curvature, as already discussed in Section 5, is sent to Eq. (27). In Fig. 11, the parabola in question is the one labelled with “no continuum estimation”. The higher n , the more accurate is the curvature. The white squares in Fig. 12 show how the curvature varies with n . Low values of n signify faster MD steps, but, at the same time, lower accuracy and, therefore, higher noise.

We now introduce an estimate of the solution. We may, for instance, assume that the fluid is Newtonian with $\mu = 2$. We may also know that the accuracy of this estimation is $\pm 50\%$ and, therefore, the real solution is bounded by the continuum profiles calculated with $\mu = 1$ and $\mu = 3$ as indicated in Fig. 11. Previously, the parabola labelled with “no continuum estimation” was calculated by standard unweighted fitting, now we can use the approximate solution and achieve a weighted fitting. Maximum weight ($w = 1$) is given to

the data in the grey area of Fig. 11. The weight of the white circles outside this zone is reduced proportionally to the square of their distance from the grey area. Other strategies are possible, but the logic is always the same: the farther the circle is from the maximum confidence zone, the smaller the weight should be. If we repeat the fitting with these weights, we obtain the parabola labelled “with continuum estimation”, which is considerable closer to n_∞ than the “no continuum estimation” one. The triangles in Fig. 12 show that the curvatures calculated with the continuum estimation are, at low n , considerably more accurate than the ones calculated without continuum estimation. Higher confidences in the continuum approximation produce higher accuracies in the curvature, as the $\mu = 2 \pm 25\%$ curve in Fig. 12 (the maximum confidence zone is now delimited by the $\mu = 1.5$ and $\mu = 2.5$ curves) indicates.

The method described here can be useful to further reduce the fluctuations. We must keep in mind, however, that it requires, at each macro-step, the calculation of two continuum solutions: one for the upper limit and the other for the lower limit of the maximum confidence zone. This, nevertheless, should not be a problem in practice for two reasons. First, we do not seek the continuum solution in the whole macroscopic domain, but only in the zones covered by the MD solver. Second, it is true that the computation of the continuum solution requires additional computational time, but usually the bottleneck of the hybrid approach lays in the microscopic rather than the macroscopic step. Therefore, if we could reduce the fluctuations by employing the macroscopic rather than the microscopic solver, we would, in general, save time.

References

- [1] P. Abgrall, N.T. Nguyen, *Nanofluidics*, Artech House Publishers, 2009.
- [2] A. Ahmed, R. Sadus, Nonequilibrium equation of state for Lennard–Jones fluids and the calculation of strain-rate dependent shear viscosity, *AIChE J.* 57 (2011) 250–258.
- [3] A. Alexiadis, S. Kassinos, Molecular simulation of water in carbon nanotubes, *Chem. Rev.* 108 (2008) 5014–5034.
- [4] A. Alexiadis, S. Kassinos, Self-diffusivity, hydrogen bonding and density of different water models in carbon nanotubes, *Mol. Simul.* 34 (2008) 671–678.
- [5] A. Alexiadis, S. Kassinos, Influence of water model and nanotube rigidity on the density of water in carbon nanotubes, *Chem. Eng. Sci.* 63 (2008) 2793–2797.
- [6] A. Alexiadis, S. Kassinos, The density of water in carbon nanotubes, *Chem. Eng. Sci.* 63 (2008) 2047–2056.
- [7] F. Alfieri, M.K. Tiwari, I. Zinovic, D. Poulikakos, T. Brunschweiler, B. Michel, 3D integrated water cooling of a composite multilayer stack of chips, *J. Heat Transfer* 132 (2010) 21402–21409.
- [8] M. Allen, D. Tildesley, *Computer Simulation of Liquids*, Clarendon Press, Oxford, 1987.
- [9] H. Akima, A new method of interpolation and smooth curve fitting based on local procedures, *J. ACM* 17 (1970) 589–602.
- [10] N. Asproulis, M. Kalweit, D. Drikakis, A hybrid molecular continuum method using point wise coupling, *Adv. Eng. Softw.* 46 (2012) 85–92.
- [11] D. Drikakis, N. Asproulis, Multi-scale computational modelling of flow and heat transfer, *Int. J. Numer. Meth. Heat Fluid Flow* 20 (2010) 517–528.
- [12] R.B. Bird, W.E. Stewart, E.N. Lightfoot, *Transport Phenomena*, second ed., John Wiley & Sons, 2007.
- [13] M.K. Borg, D.A. Lockerby, J.M. Reese, A multiscale method for micro/nano flows of high aspect ratio, *J. Comput. Phys.* 233 (2013) 400–413.
- [14] J. Cormier, J.M. Rickman, T.J. Delph, Stress calculation in atomistic simulations of perfect and imperfect solids, *J. Appl. Phys.* 89 (2001) 99–104.
- [15] W. E, W.C. Ren, E.C. Vanden-Eijnden, A general strategy for designing seamless multiscale methods, *J. Comput. Phys.* 228 (2009) 5437–5453.
- [16] N.G. Hadjiconstantinou, A.T. Patera, Heterogeneous atomistic-continuum representations for dense fluid systems, *Int. J. Mod. Phys. C* 8 (1997) 967–976.
- [17] G. Hu, D. Li, Multiscale phenomena in microfluidics and nanofluidics, *Chem. Eng. Sci.* 62 (2007) 3443–3454.
- [18] J.H. Irving, J.G. Kirkwood, The statistical mechanical theory of transport processes. IV. The equations of hydrodynamics, *J. Chem. Phys.* 18 (1950) 817–829.
- [19] M. Gad-El-Hak, *MEMS: Introduction and Fundamentals*, Taylor & Francis, 2006.
- [20] I.G. Kevrekidis, G. Samaey, Equation-free multiscale computation: algorithms and applications, *Annu. Rev. Phys. Chem.* 60 (2009) 321–344.
- [21] J. Koo, C. Kleinstreuer, Viscous dissipation effects in microtubes and microchannels, *Int. J. Heat Mass Transfer* 47 (2004) 3159–3169.
- [22] L.D. Landau, E.M. Lifshitz, *Fluid Mechanics*, Pergamon Press, New York, 1959.

- [23] T. Li, Y.G. Lv, J. Liu, Y.X. Zhou, A powerful way of cooling computer chip using liquid metal with low melting point as the cooling fluid, *Forsch. Ingenierwes.* 70 (2005) 243–251.
- [24] J. Liu, S. Chen, X. Nie, M.O. Robbins, A continuum-atomistic simulation of heat transfer in micro- and nano-flows, *J. Comput. Phys.* 227 (2007) 279–291.
- [25] T.W. Lion, R.J. Allen, Computing the local pressure in molecular dynamics simulations, *J. Phys.: Condens. Matter* 24 (2012) 284133.
- [26] J.F. Lutsko, Stress and elastic constants in anisotropic solids: molecular dynamics techniques, *J. Appl. Phys.* 64 (1988) 1152–1154.
- [27] K.M. Mohamed, A.A. Mohamad, A review of the development of hybrid atomistic-continuum methods for dense fluids, *Microfluid. Nanofluid.* 8 (2010) 283–302.
- [28] W.D. Nicholls, M.K. Borg, D.A. Lockerby, J.M. Reese, Water transport through (7,7) carbon nanotubes of different lengths using molecular dynamics, *Microfluid. Nanofluid.* 12 (2012) 257–264.
- [29] W.D. Nicholls, M.K. Borg, D.A. Lockerby, J.M. Reese, Water transport through carbon nanotubes with defects, *Mol. Simul.* 38 (2012) 781–785.
- [30] D.C. Rapaport, *The Art of Molecular Dynamics Simulation*, second ed., Cambridge University Press, 2004.
- [31] W. Ren, W. E, Heterogeneous multiscale method for the modeling of complex fluids and micro-fluidics, *J. Comput. Phys.* 204 (2005) 1–26.
- [32] B.D. Todd, D.J. Evans, P.J. Davis, Pressure tensor for inhomogeneous fluids, *Phys. Rev. E* 52 (1995) 1627–1638.
- [33] C. Trozzi, G. Ciccotti, Stationary non-equilibrium states by molecular dynamics. II. Newton's law, *Phys. Rev. A* 29 (1984) 916–925.