Visualising and controlling the flows in biomolecular systems at and between multiple scales: from atoms to hydrodynamics at different locations in time and space

Evgen Pavlov,*[a] Makoto Taiji,[c] Arturs Scukins,[a] Anton Markesteijn,[b] Sergey Karabasov,[b] and Dmitry Nerukh[a]

A novel framework for modelling biomolecular systems at multiple scales in space and time simultaneously is described. The atomistic molecular dynamics representation is smoothly connected with statistical continuum hydrodynamics description. The system behaves correctly at the limits of pure molecular dynamics (hydrodynamics) and at the intermediate regimes when the atoms move partly as atomistic particles, and at the same time follow the hydrodynamic flows. The corresponding contributions are controlled by a parameter, which is defined as an arbitrary function of space and time, thus, allowing an effective separation of the atomistic 'core' and continuum 'environment'. To fill the scale gap between the atomistic and the continuum representations our special purpose computer for molecular dynamics, MDGRAPE-4, as well as GPU-based computing were used for developing the framework. These hardware developments also include interactive molecular dynamics simulations that allow to intervene the modelling through force-feedback devices.

1 Introduction

The entirety of physical processes in complex systems can be represented as a hierarchical structure. The processes in chemical system, macromolecules, or biological objects proceed simultaneously at different temporal and spatial scales. Recently attempts have been made in building integrated models that allow carrying out computer experiments simultaneously at several scales [1–3]. This is especially important for biological objects, starting from protein molecules and ending with human organs [4,5]. This kind of models contain and exchange information between the components at different scales providing more complete picture and uncovering qualitatively new phenomena [6,7].

While statistical mechanics methods allow the description of the system as a whole using average values, the direct implementation of models with different levels of the hierarchy demands understanding the physical behaviour of materials at different levels. For example, in solid state physics interatomic interaction models [8] describe the type of bond formation and the structure of the material. From the other hand, the method of finite elements, which completely ignores internal atomistic structure, describes it as a continuum model [9].

[a] *Non-linearity and Complexity Research Group, Aston University, Birmingham, B4 7ET, UK, E-mail: e.pavlov@aston.ac.uk*
[b] *School of Engineering & Materials Science, Queen Mary, University of London, Mile End Road, London E1 4NS, UK, E-mail: s.karabasov@qmul.ac.uk*
[c] *Computational Biology Research Core, RIKEN QBiC (Quantitative Biology Center) 1-6-5 Minatojima-minamimachi, Chuo, Kobe, Hyogo 650-0047, Japan, E-mail: taiji@gsc.riken.jp*

An example of such models applied to the physics of liquids is the Landau-Lifshitz fluctuating hydrodynamics [10]. This model maps collective dynamic of continuum phase to the atomistic level. The hydrodynamic Landau-Lifshitz equations (LL-FH) are classical continuum fluid dynamics equations with added stochastic flows. They include the statistical model of temperature fluctuations around the local equilibrium. The fluctuations satisfy the equipartition theorem [11]. Combining this model with the method of molecular dynamics (MD), which represents the atomistic approach (everything is determined through interatomic interactions), allows to construct an hierarchical model. It is now possible to take into account the atomistic character of the movements of the molecules and the continuum approach for stochastic flows in fluid at the same time [12–14]. The application of this approach to the macromolecule-water system is promising for studying the properties of proteins and their environment.

Historically one of the first attempts to unify the continuum description of liquids with their atomistic representation was the Langevin equation used for the descriptions of the particle motion in continuum:

$$\frac{1}{m}f(t) = \frac{du}{dt} + \gamma u, \tag{1}$$

where u is the velocity of the particle of mass m moving under the influence of the force f and γ is the friction coefficient. The force is assumed to be random with correlations determined by the expression where it is defined as a white noise:

$$\langle f_i(t) f_j(t') \rangle = 2D\delta_{ij}\delta(t-t'), \tag{2}$$

where

$$D = m\gamma k_B T. \tag{3}$$

The spectral density [15] of the force is

$$\frac{1}{3}\langle f^2(t)\rangle_\omega = 2D. \tag{4}$$

It shows that D is a diffusion coefficient and at the same time it is the noise intensity in the Langevin equation. The equation (2) expresses the fluctuation-dissipation relationship [15].

In the hydrodynamic limit the simultaneous fluctuations at different locations in space are not correlated and considered as the white noise. To determine the fluctuations of the local stresses and heat fluxes in the case of continuous hydrodynamic environment Landau and Lifshitz suggested a modification of the classical Navier-Stokes equations in which the random variables are present. As the development of Langevin's idea (2), the stochastic terms are determined based on the fluctuation-dissipation theorem [11], which provides the balance between the fluctuations in the system and its dissipative properties. When numerically solving the fluctuation hydrodynamics equations it is important to keep this balance. If it is violated the system can either develop instability (with the dominance of stochastic flows) or the value of the fluctuations can be too low (under the dominance of dissipative flows in the system).

For a correct transition between the MD and the hydrodynamic descriptions the correspondence of the statistics of fluctuations in both representations is required. This is computationally demanding because some characteristics of the liquid are particularly slowly converging (such as, for example, the isothermal compressibility calculated using the derivative of the pressure with respect to volume). This is especially true at the stage of framework development as the equations at all levels have to be solved at the same time for benchmarking. To overcome the computational difficulties two ways are possible. The first path uses inexpensive gaming platforms, the second one implements supercomputers. We have used both in our work while developing our approach.

The widespread of computer games leads to the rapid evolution of graphics processors, the development of which is faster than the development of conventional CPUs. The main complication for molecular modelling applications is the need in software coding that requires special knowledge. Nevertheless, the trend is such that, possibly, the graphics processors will soon be one of the most attractive alternatives for cheap and relatively high performance computing (HPC). Our results reported below show high efficiency of GPU based computation in solving LL-FH equations.

The development of specialised petaflops platforms is more difficult, but it has a potential of fundamental breakthroughs in niche simulations such as molecular modelling as they can achieve timescales infeasible for any other hardware. The current trend in peta- (and exa-) flops HPC can be divided into two classes, Fig. 1. The first uses general purpose CPUs and achieves high performance by utilising a very large number of computing nodes. An example of such machine is the K-computer built in RIKEN. The second class is built around specialised accelerators where the most time consuming molecular computation is performed on the integrated chip, for example, the MDGRAPE or ANTON computers, Fig. 1. We have used both types of supercomputers for generating MD data necessary in our hybrid method development.

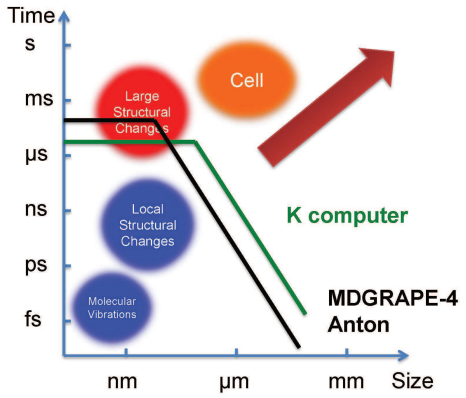

Fig. 1: Time and space scales of biomolecular processes and the limits of MD simulation feasibility

In this publication we report on the research of an international team in the framework of a G8 project 'Using next generation computers and algorithms for modelling the dynamics of large biomolecular systems' that includes the theoretical, computational, and engineering development of a framework for hybrid multiscale atomistic/hydrodynamic modelling of liquid solutions. We first describe the main idea and theoretical implementation of the approach. Then, the results on the method's numerical implementation as well as GPU and high performance specialised computer MDGRAPE are provided.

2 Multiscale hybrid method

2.1 The general idea

Our model describing both hydrodynamic (continuous) and molecular dynamic (discrete) components is based on the model of a miscible two-phase fluid. The computational domain is a double periodic box which is covered by a uniform Eulerian grid and filled with Lagrangian particles. At some regions, HD and MD, the liquid is described by purely hydrodynamic and purely Newtonian equations of motion respectively. In the hybrid domain the fluid consists of two "phases", Fig. 2. Phase 1 is a continuum water with volume fraction $s = \frac{V_1}{V}$,

where V_1 is the volume of phase 1 and V is the total volume of the domain. Phase 2 is a hydrodynamic phase that incorporates molecular particles. Its volume fraction is $(1-s)$ and it is characterised by N, the number of particles per elementary volume dV. The phase fraction $s = s(x)$ is a function of space coordinates x such that $s = 1$ in the HD domain, $s = 0$ in the MD domain, and it smoothly varies, $0 < s(x) < 1$, in the hybrid domain, Fig. 2. For each hydrodynamic cell the velocity u and the density ρ of the hydrodynamic phase are defined. The averaged value of the atomistic velocities and densities are also defined for each cell, u_p and ρ_p, Fig. 2.

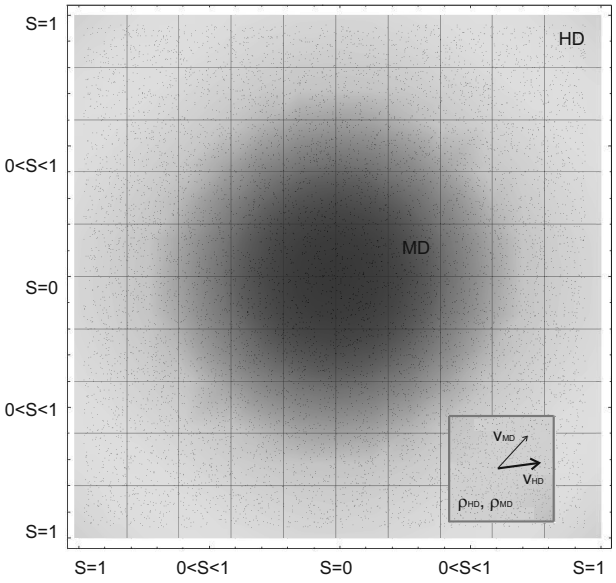

Fig. 2: Hybrid hydrodynamics – molecular dynamics model and the hydrodynamic cell (control volume)

The equations of motion are derived from the equations of conservation of total mass and momentum in each elementary Eulerian volume (control volume). The coupling between HD and MD substances is established in the HMD region by introducing the birth and death source terms for each substance, by analogy with a model of miscible two-phase flows. For determining the densities and velocities of the continuum phase the system of Landau-Lifshitz equations is solved:

$$\frac{\partial \rho}{\partial t} + \nabla(\rho \vec{u}) = 0, \tag{5}$$
$$\frac{\partial \rho_\alpha}{\partial t} + \nabla(\rho u_\alpha \vec{u}) = \nabla_\beta \left(\tilde{\Pi}_{\alpha\beta} + \Pi_{\alpha\beta}\right),$$
$$\frac{\partial \rho E}{\partial t} + \nabla(\rho E \vec{u}) = \nabla_\beta \left(\left(\Pi_{\alpha\beta} + \tilde{\Pi}_{\alpha\beta}\right) \cdot u_i\right) + \nabla(\vec{q} + \tilde{\vec{q}}).$$

Where the stress tensor consists of a deterministic part

$$\Pi_{\alpha\beta} = -\left(p - \xi \nabla \cdot \vec{u}\right)\delta_{\alpha\beta} + \eta\left(\partial_\alpha u_\beta + \partial_\beta u_\alpha - 2D^{-1}\nabla\vec{u}\cdot\delta_{\alpha\beta}\right) \tag{6}$$

and a stochastic part, a random Gaussian matrix with zero mean and the covariance given by the formula

$$\langle \tilde{\Pi}_{\alpha\beta}(r_1,t_1)\cdot\tilde{\Pi}_{\gamma\delta}(r_2,t_2)\rangle = 2k_BT\left[\eta\left(\delta_{\alpha\beta}\delta_{\alpha\gamma} + \delta_{\alpha\gamma}\delta_{\beta\delta}\right) + \left(\eta_V - \frac{2}{3}\eta\right)\times \times \delta_{\alpha\beta}\delta_{\beta\gamma}\right]\delta(r_1 - r_2)\delta(t_1 - t_2). \tag{7}$$

The stochastic stress tensor can be written explicitly[11] as

$$\tilde{\Pi}_{\alpha\beta} = \sqrt{\frac{2k_BT}{\delta t \delta V}}\left(\sqrt{2}\sqrt{\eta}\cdot\mathbf{G}^s_{\alpha\beta} + \sqrt{D}\sqrt{\xi}\frac{tr[\mathbf{G}]}{D}\mathbf{E}_{\alpha\beta}\right), \tag{8}$$

where $\mathbf{G}$ is the Gaussian random matrix with zero mean and covariance $\langle \mathbf{G}_{\alpha\beta}\mathbf{G}_{\gamma\delta}\rangle = \delta_{\alpha\gamma}\delta_{\beta\delta}$, $\mathbf{G}^s_{\alpha\beta} = \frac{\mathbf{G}_{\alpha\beta}+\mathbf{G}^T_{\alpha\beta}}{2} - \frac{tr[\mathbf{G}]}{D}\mathbf{E}_{\alpha\beta}$ are symmetric random matrices with zero trace, $\mathbf{E}_{\alpha\beta}$ is the identity matrix, $tr[\mathbf{G}]$ is the trace of the matrix. This form of correlations follows from the equipartition theorem, which relates the thermal fluctuations to temperature[11].

The averaged heat flux is

$$q_\alpha = \kappa \cdot \partial_\alpha T, \tag{9}$$

where κ is the heat conductivity coefficient, ξ and η are the shear and bulk viscosities, D is the dimensionality of the system. An additional stochastic heat flux also has zero mean and the covariance flux components are defined by

$$\langle \tilde{q}_\alpha(r_1,t_1)\cdot\tilde{q}_\beta(r_2,t_2)\rangle = 2k_B\lambda T^2\delta_{\alpha\beta}\delta(r_1 - r_2)\delta(t_1 - t_2)$$

or, explicitly,

$$\tilde{q}_\alpha = \sqrt{\frac{2k_B\lambda T^2}{\delta t \delta V}}\mathbf{G}_\alpha. \tag{10}$$

In our case the stochastic stress tensor and the stochastic heat fluxes are independent. For this reason in the following the heat flows are not considered.

The solution provides the correspondence between the averaged MD velocities of the particles in the cell, the fluctuating hydrodynamic flows, and the temperature. Our method provides the equations that give matching values of the particle velocities and the fluid velocities.

2.2 Main governing equations of the method

We use the following notations: x_i is the Cartesian coordinate component, u_i is the velocity component, ρ is the density, m is

the mass, F is the force per unit Eulerian volume $V = h^3$, N is the number of particles per unit volume, and subindex p stands for particle (MD model). Assume a summation over any index repeated twice. For simplicity here we assume that space and time is uniform everywhere in the MD and HD regions of the box. The mass conservation for the HD substance of density ρ that occupies the partial volume $0 < s < 1$ is

$$\frac{\partial(s\rho)}{\partial t} + \frac{\partial(u_i s\rho)}{\partial x_i} = J, \tag{11}$$

with an equivalent form convenient for numerical solution:

$$s(\rho(t+\tau) - \rho(t)) + \sum_p \left(s \int_t^{t+\tau} \rho u_i dt \right) dS_\beta / V = \\ - \int_t^{t+\tau} J dt, \tag{12}$$

where J and ρ represent the quantities averaged over the control volume, that is the Eulerian "bin" for MD of volume $V = V_{bin}$, area S and β is a coordinate direction. J is the HD substance birth rate due to the coupling with MD particles in the HD zone. A similar equation is formed for the MD substance that occupies the partial volume $(1-s)$:

$$\frac{\partial}{\partial t} \left((1-s) \sum_p \rho_p) + \frac{\partial}{\partial x_i} ((1-s) \sum_p \rho_p u_{ip} \right) = -J, \tag{13}$$

where $\rho_p = m_p / V$ is the density of the MD particles. Again, this differential equation is just a convenient form for the equation, which in practice is solved in the conservation form

$$(1-s) \left(\sum_p \rho_p(t+\tau) - \sum_p \rho(t) \right) + \\ \sum_\alpha \left((1-s) \int_t^{t+\tau} \sum_p \rho_p u_{\beta_p} dt \right) dS_\beta / V = - \int_t^{t+\tau} J dt, \tag{14}$$

where the flux divergence can be found from the MD

$$\sum \left((1-s) \int_t^{t+\tau} \sum_p \rho_p u_{\beta_p} dt \right) dS_\beta / V = \\ \sum_\beta \left((1-s) \left(\sum_{\beta - particles - out} \rho_p - \sum_{\beta - particles - in} \rho_p \right) \right). \tag{15}$$

By adding the same term $\frac{\partial}{\partial t}((1-s)\rho + \frac{\partial}{\partial x_i}((1-s)\rho u_i)$ to both sides of (11), it is rearranged to

$$\frac{\partial}{\partial t} p + \frac{\partial}{\partial x_i}(u_i \rho) = J + \\ + \frac{\partial}{\partial t} \left((1-s) \sum \rho) + \frac{\partial}{\partial x_i} ((1-s) \sum \rho u_i \right). \tag{16}$$

The mass birth/death function J entering the balance laws is not arbitrary but has to be determined in accordance with the modification of the MD equations. It is assumed that in the HD zone the evolution of the coordinate x_{pi} of each particle is determined by its MD velocity obtained from the Newton law that takes into account the two-phase force coupling, u_{ip}^{Newton} plus a correction. The correction is needed to constrain the velocity of the particles to the HD value in the pure HD limit $s \to 1$:

$$\frac{dx_{pi}}{dt} = u_{pi} + s(u_i - u_{ip}) + \\ s(1-s)\alpha(x) \frac{\partial}{\partial x_i} (\tilde{\rho} - \sum_p \rho_p) / \rho_p / N(t) \\ \tilde{\rho} = s\rho + (1-s) \sum_p \rho_p. \tag{17}$$

Equation (17) is equivalent to using a modified force potential in the usual MD equations (although the explicit formula for the modified potential that produces the corrected velocity field is not trivial to obtain because of the complicated particle-particle interactions involved).

The momentum equation for the HD substance follows from the conservation law:

$$\frac{\partial(s\rho u_i)}{\partial t} + \frac{\partial(u_j u_i s\rho)}{\partial x_j} = sF_i + J_2, \tag{18}$$

where J_2 is the MD interaction force. A similar conservation law for the MD particles per volume is:

$$\frac{\partial((1-s)\sum_p \rho_{pi} u_{pi})}{\partial t} + \frac{\partial((1-s)\sum_p \rho u_{pi} u_{pj})}{\partial x_j} = \\ = (1-s) \sum_p F_{pi} - J_2, \tag{19}$$

where $F_{pi} = F_{pi}^{MD} = \frac{\partial V_p}{\partial x_i}$, V_p-interpartical potintial. In the above the HD force is computed in accordance with the Landau-Lifshitz FH model

$$F_i = -\frac{\partial(\Pi_{\alpha\beta} + \tilde{\Pi}_{\alpha\beta})}{\partial x_i}. \tag{20}$$

Similarly to the mass equation, the momentum birth/death function J_2 entering the balance laws is not arbitrary but has to be determined in accordance with the modification of the MD equations, achieved by generalising the second Newton law:

$$\frac{du_{pj}^{Newton}}{dt} = (1-s)F_{pj}/\rho + \\ \frac{\partial}{\partial x_i} \left(s(1-s)\alpha(x) \sum_p u_{pj} / N(t) \frac{\partial}{\partial x_i} \left(\tilde{\rho} - \sum_p \rho_p \right) \right) / \rho_p / N(t) - \\ - \frac{\partial}{\partial x_i} \left(s\beta(x,t) \frac{\partial}{\partial x_i} \left(\tilde{\rho} \tilde{u_j} - \sum_p \rho_p u_{pj} \right) \right) / \rho_p / N(t), \tag{21}$$

where the density is $\tilde{\rho} = s\rho + (1-s)\sum\rho_p$, and the velocity is $\tilde{u_i} = \left[s\rho u_i + (1-s)\sum_p \rho_p u_{pi}\right]/\tilde{\rho}$, and $\beta(x,t) = \beta(x)$, for $N(t) > 0$, otherwise $\beta(x,t) = 0$.

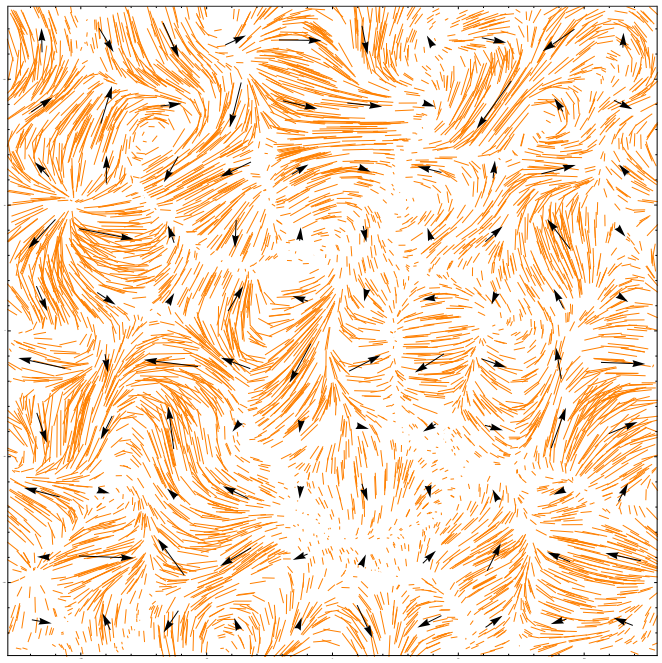

Fig. 3: Velocity fields of MD (grey) and hydrodynamic (black) phases for $s = 0.6$

3 Results

3.1 Results on the hybrid method

For computer simulations a system of 40k two-dimensional particles that interact via the Lennard-Jones potential was chosen. The simulation area was a square with periodic boundary conditions [16]. Temperature and density corresponded to water density at 122.4K. The region was divided into 100 hydrodynamic cells, each of which had the value of the hydrodynamic density and temperature derived from the Landau-Lifshitz FH model. Purely MD calculations were used for equilibration, after which the modified leapfrog formulas (21) and (17) were used for integration. The coupling between the continuum phase (FH) and the MD phase was performed at every 100 MD step (100 MD steps is equal to 1 FH step). For several values of the model parameters s, α, and β the simulations of the hybrid LL-FH/MD model have been performed. The parameters α and β were chosen equal to 10000 for efficient coupling between the phases.

The coupling parameter s was varied from zero to one. Fig. 3 and 4 show a typical behaviour of the continuum phase and the molecular dynamics phase velocities u_i and

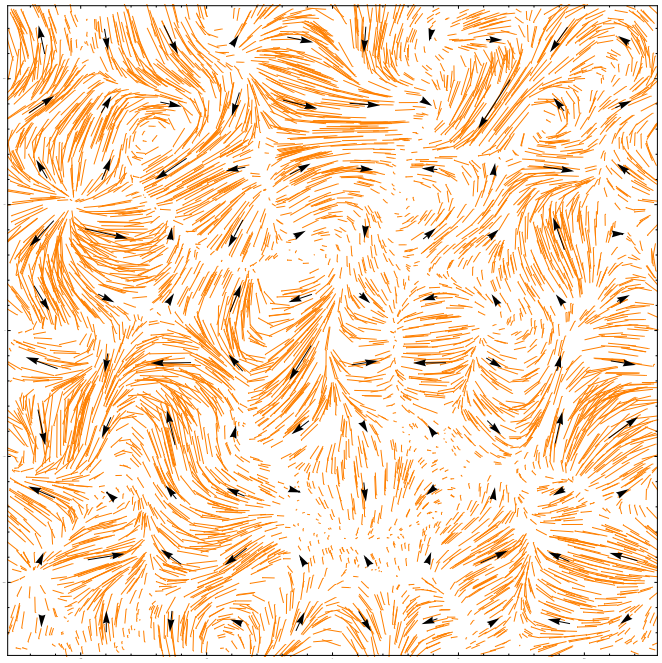

Fig. 4: Same as in Fig. 3 for $s = 0.8$

$\sum_p u_{pi}/N(t)$. Also, the curl of the velocity fields is shown in Fig. 5 and 6. The density fields are shown in Fig. 7 and 8. In all cases the MD and FH phases follow each other. The best correspondence is achieved for $s = 0.8$, as expected, since for this case the MD particles act as almost passive traces that follow the hydrodynamic flows, whereas for $s = 0.6$ discrepancies between the fields are present.

The solutions for the two phases are driven to each other and the standard deviation of the main variables converge to the reference values of the pure MD solution, which is the same as that of the pure LL-FH solution. It should be stressed that the comparisons of the field values of the hybrid method with those of pure MD (FH) are meaningless because the solutions are stochastic (chaotic) and particular realisations of the fields are different. A meaningful comparison is possible in the statistics of the fluctuations of the field values, which do match as described above.

3.2 High performance molecular modelling: MD-GRAPE

In multiscale simulations the computational speed is limited by high-resolution parts due to the large differences in time and space scales. The basic time step of an all-atom MD simulation is 1-2 fs, while those of hydrodynamics is of the order of hundreds of picoseconds. Thus, we need to fill the gap of several orders of magnitude in time scale. We have done this by using HPC.

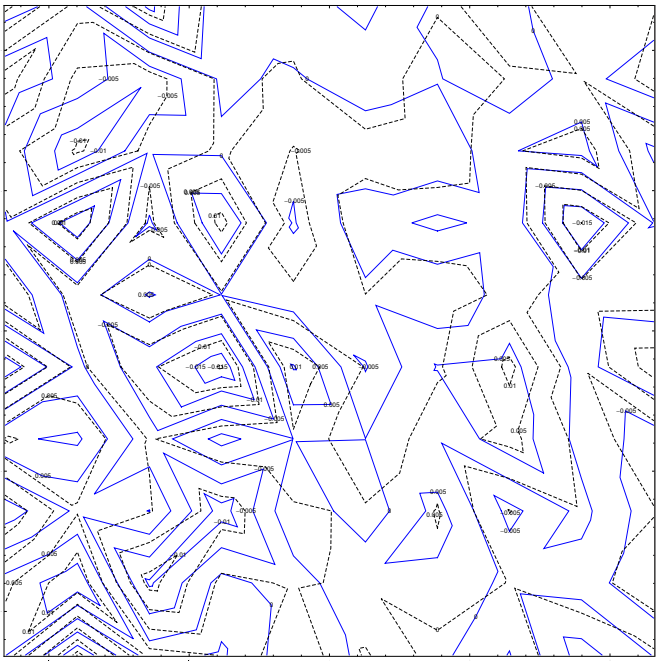

Fig. 5: Curl of the velocity fields of MD (solid line) and hydrodynamic (dashed line) phases for $s = 0.6$

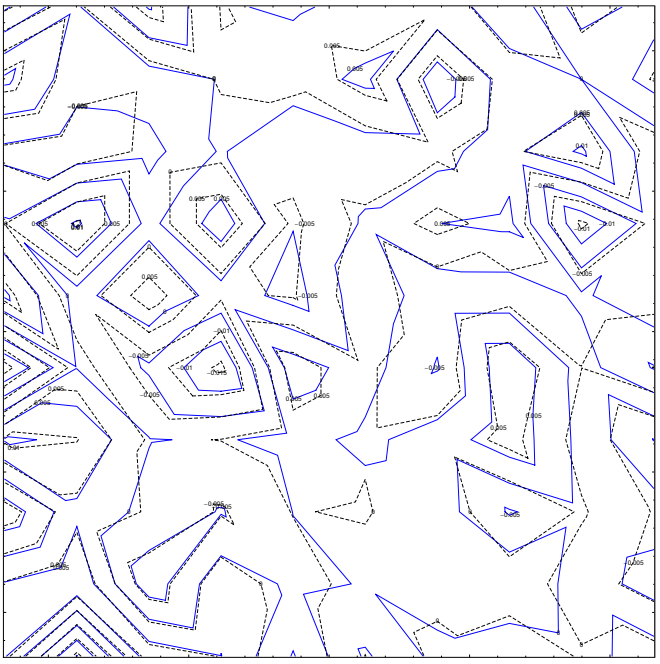

Fig. 6: Same as in Fig. 5 for $s = 0.8$

The general CPU based supercomputers, such as the K-computer, are most effective for simulating large molecular systems comprising millions of atoms. In this case the scaling of the computation speed with the number of CPUs is very good, Fig. 9. In contrast, the specialised computers allow faster calculations for smaller systems, of the size of several thousands of atoms, Fig. 1. We have developed such special-purpose computer systems for MD simulations named MDGRAPE [17], Fig. 10. The MDGRAPE computers were accelerators of non-bonded force calculations.

Recently, the ANTON machine, which is also a special-purpose computer for MD simulations, has been developed by D. E. Shaw research [18]. They integrated the non-bonded force accelerators, general-purpose processor cores, memories, and network interfaces in a single System-on-Chip (SoC). The integration minimizes the latencies between these computing elements. As a result, it enables 100 times better performance for small-scale systems than normal PC clusters and the high performance of the full system can be achieved for systems as small as 10K atoms. Thus, it enables 100 ps MD simulation per second for these systems.

The ability of enabling fast simulations of small systems is especially useful for protein MD simulations as well as multiscale simulations. For this purpose we are currently developing the fourth-generation machine, MDGRAPE-4, Fig. 11. The MDGRAPE-4 also aims at achieving fast simulations of small systems. It is also based on the SoC technology, similar to ANTON. The target performance is 100 ps simulation per second for a system with 100K atoms. Currently, we have almost finished the evaluation of the system board. We plan to finish the system in 2014 when the software development will be finalised. The latter will be based on GROMACS and contain C library set for accessing the GP core.

The calculation speed is critical for visual interaction with the simulated systems since the time of data processing in a human brain and a visual system is of the order of 10 ms. Thus, to enable a real-time interactive simulations, we need the performance of more than 30 frames per second and more than 300 simulation steps per second. We have developed such a system using the previous generation machine MDGRAPE-3, Fig. 12. In this system, only a few layers of water molecules can be treated due to the performance limitation. A typical speed is $\approx$ 15 frames per second for a $\approx$ 150 residue globular protein. The multiscale simulation coupling MD on the MDGRAPE-4 and FH on conventional CPU/GPU machines will enable smoother and more precise visual simulations of solvated biomolecules.

3.3 GPU results

Another branch of HPC implementation of our framework uses GPU based calculations. Because of the nature of stochastic simulations, it is the statistical behaviour of the LLNS equations which is relevant for the model rather than an instantaneous solution. For statistical convergence of low fre-

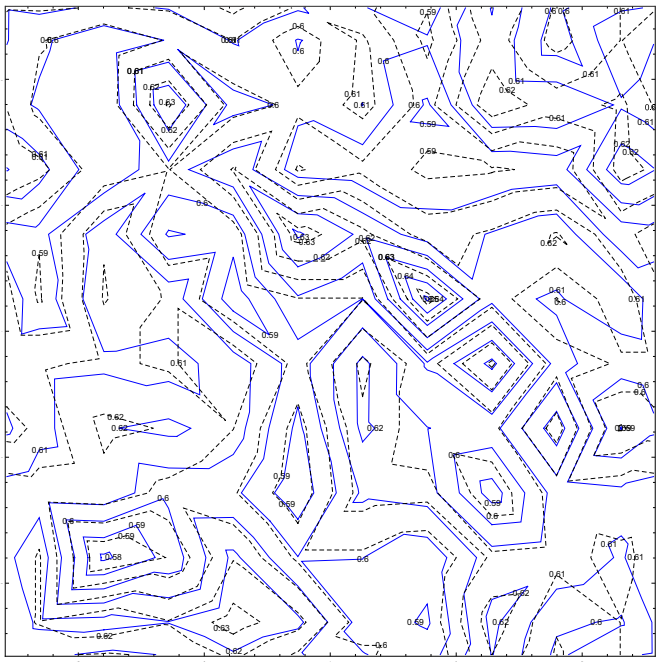

Fig. 7: Density fields of MD (solid line) and hydrodynamic (dashed line) phases for $s = 0.6$

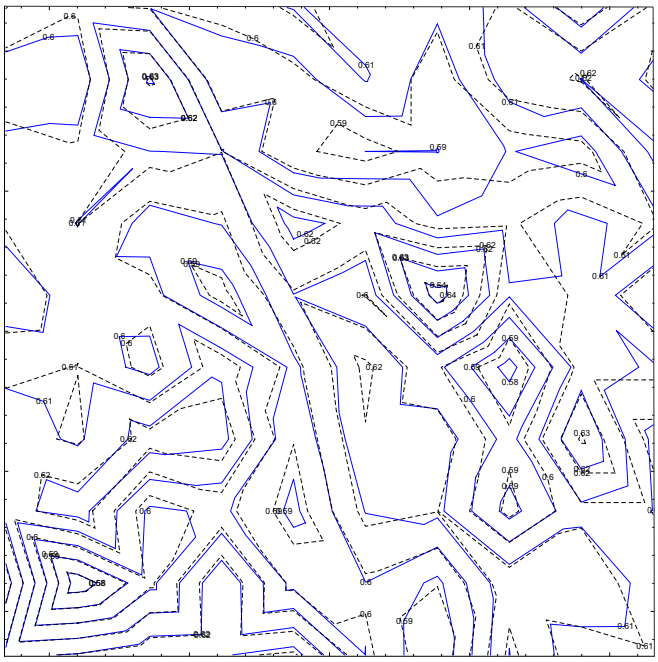

Fig. 8: Same as in Fig. 7 for $s = 0.8$

quencies typical of the hydrodynamics fluctuations, one needs a sufficiently long simulation time. For example, the solution of stochastic diffusion problem converges as $\frac{1}{\sqrt{N_t}}$, where N_t is the number of time steps of the simulation, which is typically of order of several millions to get the error in fluctuations down to less than a few percent.

For three-dimensional problems, this leads to considerable computational times in case it is run in a serial manner based on single computational unit (CPU). On the other hand, the LL-NS Fluctuating Hydrodynamics equations are well suited for an implementation in NVidia's CUDA. The latter allows one to off-load the computations on to the Graphical Processing Units (GPU) with performing massively parallel computations based on hundreds of computational threads.

The basic idea behind computations on the GPU, using the terminology of CUDA, are based on the fact that certain bits of code (kernels) can be executed simultaneously in several blocks each containing hundreds of threads. The key for optimisation is to limit the request for the same memory space, resulting in memory collisions and therefore high latency. In the case of the LL-NS equations, each block of cells (e.g. 16x16) can be evaluated independently, where only the single layer of boundary cells need to request memory space shared with other blocks. This is similar to simple block domain decomposition. Besides the different approach how to the access of the computational arrays, the (C++) code itself needs little change.

The GPU used in the current simulations of liquid argon at equilibrium conditions is the NVidia Tesla C2075. Figure 13 shows the amount of time it takes to compute one cycle of the two-time-level Central Leapfrog algorithm for three-dimensional LL-NS equations versus the number of cells of the computational domain.

The dependency of the amount of workload goes up linearly with the amount of cells, i.e. a linear scaling if workload balance is achieved. It is also interesting to compare this scaling with the workload based on a single CPU. Figure 13(b) shows how the GPU computation times compare to the computational times based on a single core of an Intel Xeon E5-2609 processor. There are two "regimes" on the CPU versus GPU comparison plot.

In the first part of the curve 13 corresponding to relatively small job sizes, the curve rises steeply that indicates a sharp growth of efficiency when only a fraction of the GPU threads are fully utilised and their competition for shared memory is low. After all GPU threads are fully engaged, the rate of the curve growth stagnates. The latter is likely to be caused by the fact that the system is reaching its limit of available RAM (GPU dedicated 5GB versus CPU shared with system 6GB). Nevertheless, as follows from this comparison graph, that the GPU can do the task up to 300 times faster compared to one CPU core, or at best about 75 times faster in comparison with the four-core CPU processor.

Because of the significant speedup due to GPU implementation, the LL-NS fluctuating hydrodynamics simulations per-

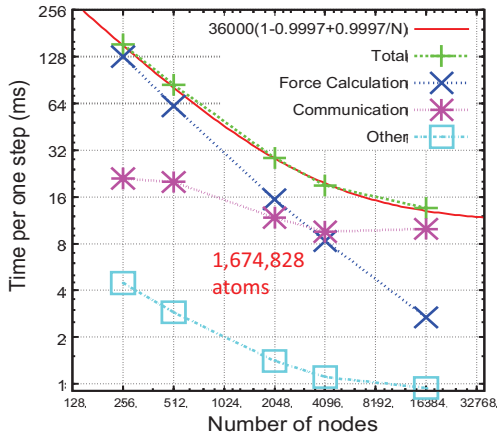

Fig. 9: Scaling with the number of computation nodes for MD simulations using the K-computer at RIKEN

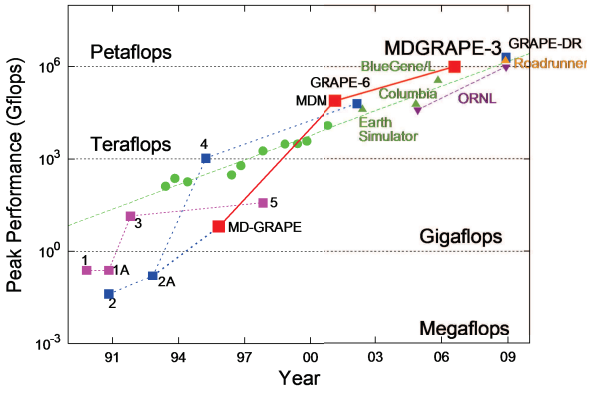

Fig. 10: The development of the GRAPE family accelerators (Gordon Bell Prizes won in 1995, 1996, 1999, 2000 (double), 2001, 2003, and 2006)

formed for this article took a few hours rather than a week on the grids up to 10 million computational cells.

4 Conclusions

A theoretical framework that allows a correct hybridisation of the atomistic (molecular dynamics) and continuum (hydrodynamics) representations of a liquid system is described. The velocity and density fields of such hybrid systems are calculated which demonstrate that the atoms can follow the continuous flows without violating the conservation laws. Importantly, the framework does not require any artificial repulsive walls or forces that separate the representations. Instead, the contribution of each representation is governed by an arbitrary function of space and time. This allows the definition of different domains, which can also move, containing different degree of 'atomisation'. This can be used in simulating, for example, a protein surrounded by water that gradually changes its representation from purely atomistic in the vicinity of the protein to purely continuous far from it.

For implementing the framework, a special purpose accelerator, MDGRAPE, has been used. The latest version of it, MDGRAPE-4, can achieve the 100 ps per day performance for a 100K atoms system. The details of the system are described and compared to other high performance computer systems used for molecular modelling.

An implementation of the continuum fluctuating hydrodynamics solver using NVidia Tesla C2075 GPU is described. A very effective speed up in computation is demonstrated.

Our current research is centred around the applications of the described framework and its HPC implementation to water-protein realistic systems. We expect the first results of this work to be published by spring 2014.

Acknowledgement. This work has been supported in the framework of the G8 Research Councils Initiative on Multilateral Research Funding (Engineering and Physical Sciences Research Council Grant No. EP/J004308/1)

References

1 P. Espanol, *Europhysics Letter*, 1997, **40**, 631.
2 N. Pasquale, D. Marchisio and P. Carbone, *Journal of Chemical Physics*, 2012, **137**, 164111–164119.
3 S. Fritsch, S. Poblete, C. Junghans, G. Ciccotti, L. D. Site and K. Kremer, *Phys. Rev. Lett.*, 2012, **108**, 170602.
4 D. Nerukh and S. Karabasov, *J. Phys. Chem. Lett.*, 2013, **4**, 815–819.
5 H. Frauenfelder, G. Chen, J. Berendzen, P. Fenimore, H. Jansson, B. McMahon, I. Stroe, J. Swenson and R. Young, *Proc. Natl. Acad. Sci. USA*, 2009, **106**, 5129–5134.
6 *Multiscale Modeling and Simulation of Composite Materials and Structures*, ed. W. Young, H. David and R. Talreja, Springer, Chichester, 6th edn., 2007.
7 P. Ajayan, L. Schadler and P. Braun, *Nanocomposite Science and Technology*, 222Wiley, New York, 3rd edn., 2006.
8 W. Harrison, *Pseudopotentials in the theory of metals*, W.A. Benjamin (New York), 1966.
9 G. Dhatt, E. Lefrancois and G. Touzot, *Finite Element Method*, Wiley, 2012.
10 G. K. Batchelor, *An Introduction to Fluid Dynamics*, Cambridge University Press, 1967.
11 L. Landau and E. Lifshitz, *Statistical Physics*, Oxford: Pergamon, 1980.

12 A. Asproulis, M. Kalweit and D. Drikakis, *Advances in Engineering Software*, 2012, **46**, 85–92.
13 G. Fabritiis, R. Delgado-Buscalioni and P. Coveney, *Physical Review Letters*, 2006, **97**, 134501.
14 G. Fabritiis, M. Serrano, R. Delgado-Buscalioni and P. Coveney, *Physical Review E*, 2007, **75**, 026307.
15 Y. Klimontovich, *Uspekhi Fizicheskikh Nauk*, 1987, **151**, 309–332.
16 D. Rapaport, *The Art of Molecular Dynamics Simulation*, Cambridge University Press, 2004.
17 J. Makino and M. Taiji, *Scientific Simulations with Special-Purpose Computers*, John Wiley & Sons, 1997.
18 D. E. Shaw, M. M. Deneroff, R. O. Dror, J. S. Kuskin, R. H. Larson, J. K. Salmon, C. Young, B. Batson, K. J. Bowers, J. C. Chao, M. P. Eastwood, J. Gagliardo, J. P. Grossman, C. R. Ho, D. J. Ierardi, I. Kolossváry, J. L. Klepeis, T. Layman, C. McLeavey, M. A. Moraes, R. Mueller, E. C. Priest, Y. Shan, J. Spengler, M. Theobald, B. Towles and S. C. Wang, *Commun. ACM*, 2008, **51**, 91–97.

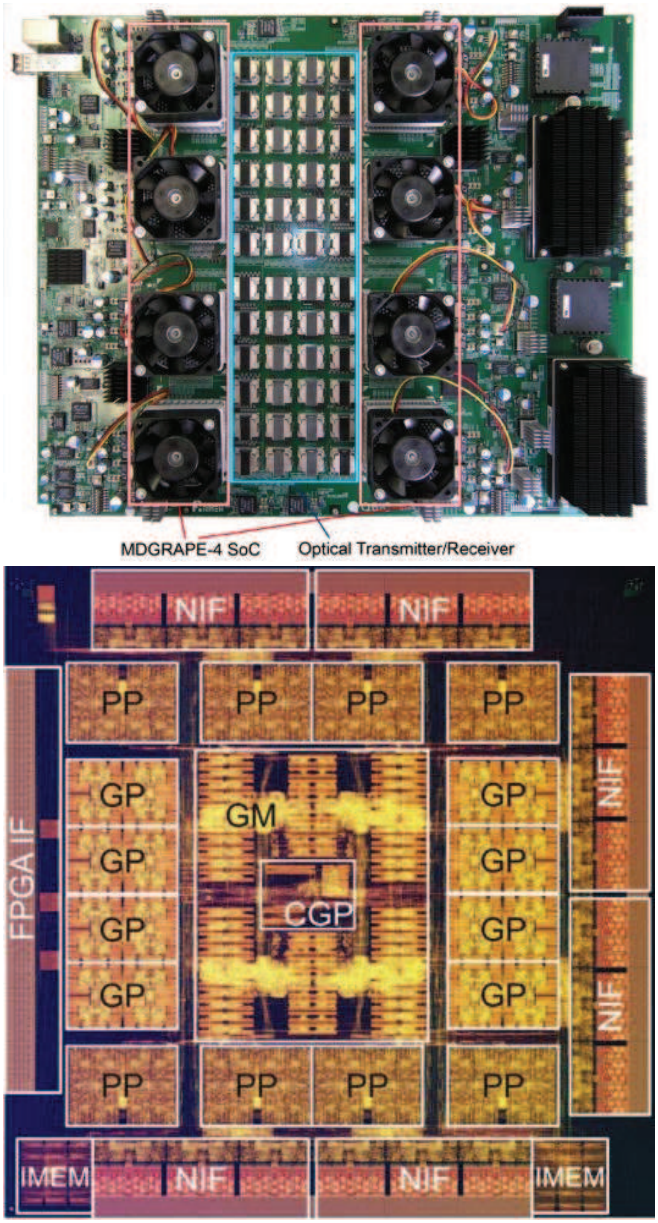

Fig. 11: The MDGRAPE-4 system board and System-on-Chip. The board has 8 SoCs and 48 optical transmitter/receiver for networking. The SoC has 8 nonbond force calculation modules (PP), 8 general-purpose processor modules (GP) each of which has 8 cores, memories (GM), and network interfaces (NIF)

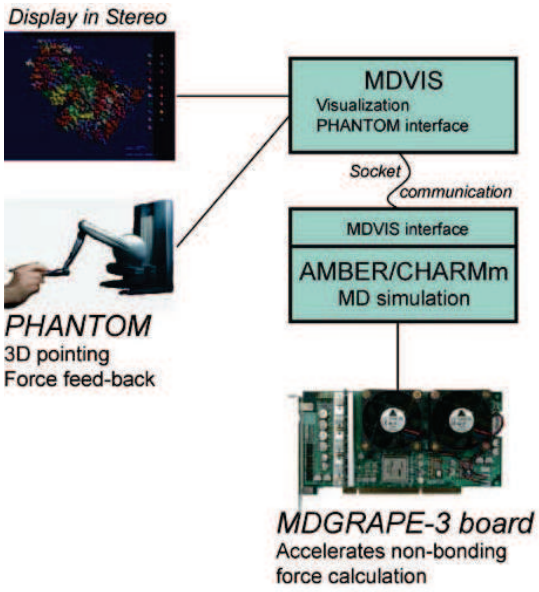

Fig. 12: Interactive MD simulation system using MDGRAPE-3. User can intervene simulations using the 3D force-feedbacked joystick

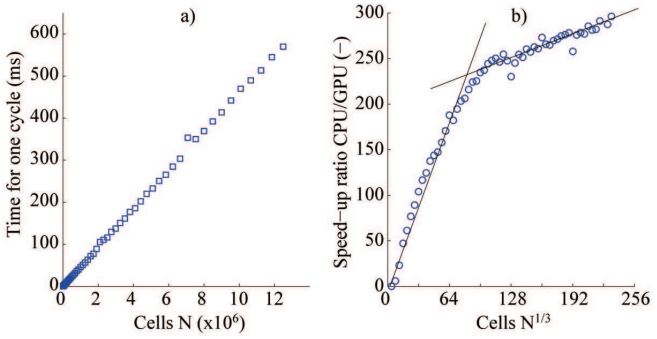

Fig. 13: a) The amount of time one cycle takes on the GPU versus the number of cells (N). b) The ratio between the time it takes to execute one cycle on the CPU versus the time on the GPU versus the number of cells in one direction of the cubed domain.