Multi-resolution dimer models in heat baths with short-range and long-range interactions

Ravinda S. Gunaratne,[1, a)] Daniel B. Wilson,[1, b)] Mark B. Flegg,[2, c)] and Radek Erban[1, d)]

[1)]*Mathematical Institute, University of Oxford, Radcliffe Observatory Quarter, Woodstock Road, Oxford, OX2 6GG, United Kingdom*

[2)]*School of Mathematical Sciences, Monash University, 9 Rainforest walk, Clayton campus, Victoria 3168, Australia*

(Dated: 17 February 2019)

This work investigates multi-resolution methodologies for simulating dimer models. The solvent particles which make up the heat bath interact with the monomers of the dimer either through direct collisions (short-range) or through harmonic springs (long-range). Two types of multi-resolution methodologies are considered in detail: (a) describing parts of the solvent far away from the dimer by a coarser approach; (b) describing each monomer of the dimer by using a model with different level of resolution. These methodologies are then utilised to investigate the effect of a shared heat bath versus two uncoupled heat baths, one for each monomer. Furthermore the validity of the multi-resolution methods is discussed by comparison to dynamics of macroscopic Langevin equations.

I. INTRODUCTION

Molecular dynamics (MD) approaches, based on the rules of classical mechanics, have been used to study the behaviour of complex biomolecules in biological applications[1,2]. They are written in terms of the positions and velocities of particles, representing either individual atoms or groups of atoms, describing parts of a biomolecule[3–6]. Inter-particle forces in MD models include combinations of short-range and long-range interactions[7,8]. In all-atom MD models, a common example of short-range forces are interactions described by the Lennard-Jones potential[9,10], while Coulomb forces provide an example of long-range forces[7]. Considering coarse-grained or caricature MD models, short-range interaction models include systems when particles only interact through direct collisions[11–14], while long-range interactions also include models, where particles interact through harmonic-springs[15,16]. Once the inter-particle interactions are specified, MD describes the time evolution of the model as a system of ordinary or stochastic differential equations for the positions of particles, which can also be subject to algebraic constraints, representing bonds between atoms or fixed internal structures of a biomolecule[2,17,18].

Biologically relevant simulations have to be done in aqueous solutions. A number of water models have been developed in the literature to use in all-atom MD simulations, including commonly used three-site (SPC/E, TIP3P) models[19,20]. In coarse-grained MD models, water is often treated with the same level of coarse-graining as other molecules in the system. For example, four water molecules are combined into a single coarse-grained water bead in the Martini model[3], while Wat Four water model[6] uses four linked beads placed at the corners of a tetrahedron to collectively represent 11 water molecules. In this paper, we consider two theoretical heat baths which enable more analytical progress than solvent models based on all-atom or coarse-grained water models. In both cases, the convergence to the Langevin description of the solute particle can be established in a certain limit[11–16]. Our solute particle will also be treated with the same level of simplicity and described as a simple dimer molecule consisting of two monomers (beads) connected by a spring.

Multi-resolution (hybrid) methods use detailed and coarse-grained simulations in different parts of the simulation domain during the same dynamic simulation[21–24]. Such methods have been developed in different application areas and at different spatial and temporal scales in the literature, including dual-resolution approaches AdResS and H-AdResS for all-atom MD simulations[25–29], methods for coupling Brownian dynamics approaches with lattice-based stochastic reaction-diffusion models[30–32] or methods which make use of continuum mean-field equations for the macroscopic component of the simulation[33–35].

In some multi-resolution MD approaches, the region of high resolution moves together with the large microscopic structure of interest so that the high resolution model is always used for the whole considered structure, which can range in size from a single biomolecule (a protein or a DNA in solution[27,28]) to virus-like particles[36,37]. The structure of interest is placed in the centre of the simulation domain and it is solvated using a detailed atomistic MD water in its immediate neighbourhood, which is coupled with a coarse-grained water description in the rest of the computational domain.

Another type of multi-resolution modelling is used for modelling of macromolecules where a detailed model of an important part of a macromolecule is coupled with

---

[a)]Electronic mail: gunaratner@maths.ox.ac.uk
[b)]Electronic mail: wilsond@maths.ox.ac.uk
[c)]Electronic mail: mark.flegg@monash.edu
[d)]Electronic mail: erban@maths.ox.ac.uk

a coarser model of the rest of the macromolecule. For example, atomistic detail of the active part of an enzyme has been coupled with a coarser model of the rest of the protein[38], different resolutions have been used in bead-spring modelling of DNA[39,40] or for modelling of polymer melts[41,42].

In this paper, we study both multi-resolution approaches using a simple dimer model consisting of two monomers (beads) connected by a spring. Similar models, where a macromolecule is described as several beads, representing parts of the simulated biomolecule, connected by springs, have been obtained in the literature using the method of ultra-coarse-graining[43]. Thus our dimer model can be considered as a caricature of an ultra-coarse-grained model of a macromolecule. We study its behaviour in two theoretical heat baths. Our investigation focuses on multi-resolution (multiscale) descriptions of the solvent which can be described at the microscopic level of individual solvent molecules or at the macroscopic (dimer) level with the introduction of extrinsic random thermal forces on the monomers. We present models of the same dimer with various multi-resolution descriptions for the solvent and highlight the conditions and reasons, when and why, different model approximations of the solvent may be made in simulations.

Our paper is organized as follows. In Section II, we introduce the macroscopic dimer model with a macroscopic description for solvent forces. This macroscopic model is fully described by Langevin equations. The Langevin macroscopic model is commonly used in simulation due to ease of implementation and analysis. We discuss in Section II the properties of this description with the intent to use these properties as benchmarks against which to compare microscopic and multi-resolution solvent models for the same dimer. Two theoretical microscopic approaches to model the solvent are introduced and studied through multi-resolution (simultaneous microscopic and macroscopic coupled) modelling in Sections III and IV. One of them is based on (very) short-range interactions, as heat bath particles only interact with the dimer on contact. The other one is at the opposite extreme, as it is based on (very) long-range interactions, where the heat bath is modelled as a system of many harmonic oscillators.

II. THE DIMER MODEL

In this section we will talk exclusively about the construction of the model for the dimer which will be used throughout this manuscript. In doing so, we describe the solvent at the macroscopic level as an extrinsically added random force. The result will be a set of Langevin equations. Throughout the manuscript we will modify the treatment of the solvent forces at various scales and hybrid resolutions but the underlying dimer model will be the same.

We consider a model of a dimer which is described by positions of its two monomers, denoted by $\mathbf{X}_1 = [X_{1;1}, X_{1;2}, X_{1;3}]$ and $\mathbf{X}_2 = [X_{2;1}, X_{2;2}, X_{2;3}]$, respectively. Each monomer has the same mass, M. We denote by $\mathbf{R}$ the vector describing the separation between the monomers, i.e. $\mathbf{R} = \mathbf{X}_2 - \mathbf{X}_1$, and by R its magnitude $R = |\mathbf{R}|$. The interaction between monomers is given in terms of the potential $\Phi \equiv \Phi(R) : [0, \infty) \to \mathbb{R}$, which generates a force on each of the monomers with magnitude $\Phi'(R)$.

When the dimer is placed into a heat bath, there are additional forces on the two monomers caused by interactions with solvent molecules. The solvent forces can be modelled in a number of different ways and at various scales. In this manuscript, we consider two classes of models to describe the solvent-dimer interactions. The first, presented in Section III, models the solvent as a bath of point particles which collide with the monomers and elastic collisions (short-range interactions) contribute to the generation of the forces. In the second case, described in Section IV, solvent molecules are point particles which oscillate around and interact at a distance (through long-range interactions) with the monomers. The solvent-dimer interactions are the sum of harmonic oscillatory forces acting on each of the monomers. Importantly, both descriptions under suitable assumptions lead to a macroscopic description of the dimer given by the following set of Langevin equations

$$d\mathbf{X}_1 = \mathbf{V}_1 \, dt, \tag{1}$$

$$d\mathbf{V}_1 = \frac{\Phi'(R)}{M} \frac{\mathbf{R}}{R} \, dt - \gamma \mathbf{V}_1 \, dt + \gamma \sqrt{2D} \, d\mathbf{W}_1, \tag{2}$$

$$d\mathbf{X}_2 = \mathbf{V}_2 \, dt, \tag{3}$$

$$d\mathbf{V}_2 = -\frac{\Phi'(R)}{M} \frac{\mathbf{R}}{R} \, dt - \gamma \mathbf{V}_2 \, dt + \gamma \sqrt{2D} \, d\mathbf{W}_2, \tag{4}$$

where $\mathbf{V}_1 = [V_{1;1}, V_{1;2}, V_{1;3}]$ and $\mathbf{V}_2 = [V_{2;1}, V_{2;2}, V_{2;3}]$ are velocities of the first and second monomer, respectively, $\mathbf{W}_1$ and $\mathbf{W}_2$ are three-dimensional vectors of independent Wiener processes, D is a diffusion coefficient and γ is a friction coefficient, with dimension $[\gamma] = [\text{time}]^{-1}$.

System (1)–(4) provides a macroscopic model of the dimer, which we compare with microscopic (or multi-resolution) MD simulations which explicitly model the solvent. Its validity for different MD models can be tested by comparing values of different dimer's statistics at equilibrium, including its expected length L_d, dimer velocity autocorrelation function $C_d(\tau)$ and dimer diffusion constant D_d, defined by

$$L_d = \lim_{t \to \infty} \langle R \rangle,$$

$$C_d(\tau) = \lim_{t \to \infty} \frac{1}{3} \left\langle \overline{\mathbf{V}}(t + \tau) \cdot \overline{\mathbf{V}}(t) \right\rangle, \tag{5}$$

$$D_d = \lim_{t \to \infty} \frac{1}{6t} \left\langle \left(\overline{\mathbf{X}}(t) - \overline{\mathbf{X}}(0) \right)^2 \right\rangle,$$

where $\overline{\mathbf{X}} = (\mathbf{X}_1 + \mathbf{X}_2)/2$ is the centre of mass of the dimer and $\overline{\mathbf{V}} = (\mathbf{V}_1 + \mathbf{V}_2)/2$ is its velocity. These quantities can be obtained analytically for our macroscopic model (1)–(4) as follows. Adding equation (2) and equation (4) and

noting that the sum of two independent Wiener processes is another Wiener processes $\mathbf{W}$ with an infinitesimal variance which is the sum of the variances of the original two processes, we obtain an Ornstein-Uhlenbeck process for $\overline{\mathbf{V}}$ in the following form

$$d\overline{\mathbf{V}} = -\gamma\overline{\mathbf{V}}\,dt + \gamma\sqrt{D}\,d\mathbf{W}.$$

Therefore, we have

$$C_d(\tau) = \frac{D\gamma}{2}\exp[-\gamma\tau]. \quad (6)$$

Integrating over τ, we deduce

$$D_d = \int_0^\infty C_d(\tau)\,d\tau = \frac{D}{2}. \quad (7)$$

Taking the difference of equation (4) minus equation (2), implementing the over-damped assumption (where γ is large) and combining the independent Weiner processes into a single Weiner processes $\mathbf{W}$ gives

$$d\mathbf{R} = -\frac{2\Phi'(R)}{M\gamma}\frac{\mathbf{R}}{R}\,dt + 2\sqrt{D}\,d\mathbf{W}.$$

The stationary distribution corresponding to this process is proportional to $\exp[-\Phi(R)/(MD\gamma)]$. Normalizing, we find the distribution of dimer lengths equal to

$$\varrho(R) = \frac{\exp\left[-\frac{\Phi(R)}{MD\gamma}\right]}{4\pi\int_0^\infty r^2\exp\left[-\frac{\Phi(r)}{MD\gamma}\right]dr}.$$

In the simulations that follow in this manuscript, we shall be assuming the dimer potential acts like a linear spring with a rest length of ℓ_0 and a spring constant of k between the two monomers. That is, we shall assume

$$\Phi(R) = \frac{k(R-\ell_0)^2}{2}. \quad (8)$$

Each monomer within the dimer is representing a half of a molecule of interest and the value of the spring constant indicates the flexibility in which the molecule can change its shape. In this paper, we consider the parameter regime where the spring constant k is sufficiently large so that the dimer has a well-defined structure. In the limit of large k, we have $\varepsilon = MD\gamma/(k\ell_0^2) \ll 1$. Then, L_d can be calculated as

$$L_d \approx \ell_0\left(1 + \frac{2MD\gamma}{k\ell_0^2}\right), \quad (9)$$

which is valid up to the first order in ε. In particular, the presence of heat baths extends the dimer from its rest length on average. In the following two sections, we study two theoretical MD models, where we use equations (6), (7) and (9) to compare the macroscopic theory with the results obtained by MD simulations.

III. SHORT-RANGE INTERACTION HEAT BATH

We describe the two monomers as balls with radius r_0 and mass M which interact with point solvent particles when they collide with them. In particular, this is a theoretical model of a (very) short-range interaction heat bath. Between collisions, monomers follow Newton's second law of motion in the form

$$M\frac{d\mathbf{V}_1}{dt} = \Phi'(R)\frac{\mathbf{R}}{R}, \quad (10)$$

$$M\frac{d\mathbf{V}_2}{dt} = -\Phi'(R)\frac{\mathbf{R}}{R}, \quad (11)$$

where, following our notation introduced in Section II, positions and velocities of the monomers are denoted by $\mathbf{X}_i$ and $\mathbf{V}_i$, respectively, and $\mathbf{R} = \mathbf{X}_2 - \mathbf{X}_1$.

Our short-range interaction heat bath is described in terms of positions $\mathbf{x}_i^j$ and velocities $\mathbf{v}_i^j$, of heat bath particles, where $i = 1, 2$ is the monomer number and $j = 1, 2, 3, \ldots$, is the number of the heat bath particle. Notice that this formulation allows us to consider two important cases: (a) each monomer has its own heat bath; (b) a single heat bath is shared by both monomers. By comparing our simulation results in cases (a) and (b), we can explicitly investigate whether there are any significant hydrodynamic interactions between the monomers. In the case (b), we simplify our notation by describing particles of the single heat bath by

$$\mathbf{x}^j = \mathbf{x}_1^j = \mathbf{x}_2^j, \quad \text{and} \quad \mathbf{v}^j = \mathbf{v}_1^j = \mathbf{v}_2^j. \quad (12)$$

In both cases (a) and (b), we assume that all heat bath particles have the same mass, m, and define (dimensionless) parameter μ by

$$\mu = \frac{M}{m}.$$

We are interested in the parameter regime where $\mu \gg 1$. Our MD model is based on elastic collisions of heavy monomers (balls with mass M and radius r_0) with point heat bath particles with masses m. We assume that the collisions are without friction, then conservation of momentum and energy yields the following formulae for post-collision velocities[12]

$$\widetilde{\mathbf{V}}_i = [\mathbf{V}_i]^\| + \frac{\mu-1}{\mu+1}[\mathbf{V}_i]^\perp + \frac{2}{\mu+1}\left[\mathbf{v}_i^j\right]^\perp, \quad (13)$$

$$\widetilde{\mathbf{v}}_i^j = \left[\mathbf{v}_i^j\right]^\| + \frac{1-\mu}{\mu+1}\left[\mathbf{v}_i^j\right]^\perp + \frac{2\mu}{\mu+1}[\mathbf{V}_i]^\perp, \quad (14)$$

where $\mathbf{v}_i^j$ is the velocity of the heat bath particle which collided with the i-th monomer, tildes denote post-collision velocities, superscripts $\perp$ denote projections of velocities on the line through the centre of the monomer and the collision point on its surface, and superscripts $\|$ denote tangential components.

Heat bath models based on elastic collisisions (13)–(14) have been studied by a number of authors[11–14]. Consider

a single monomer in infinite domain $\mathbb{R}^3$, and let the heat bath consist of an infinite number of particles with positions distributed according to the spatial Poisson process with density

$$\lambda_\mu = \frac{3}{8r_0^2}\sqrt{\frac{(\mu+1)\gamma}{2\pi D}}. \tag{15}$$

This means that the number of points in a subset Ω of $\mathbb{R}^3$ has its probability mass function given by the Poisson distribution with mean $\lambda_\mu|\Omega|$, where $|\Omega|$ is the volume of Ω. Let the velocities of the heat bath particles be distributed according to the Maxwell-Boltzmann distribution

$$f_\mu(\mathbf{v}) = \frac{1}{\sigma_\mu^3(2\pi)^{3/2}}\exp\left[-\frac{v_1^2+v_2^2+v_3^2}{2\sigma_\mu^2}\right], \tag{16}$$

where $\mathbf{v} = [v_1, v_2, v_3]$ and

$$\sigma_\mu = \sqrt{(\mu+1)D\gamma}. \tag{17}$$

Then the monomer's behaviour is known to converge to the Langevin dynamics[12,14]. In particular, if we consider that each monomer has its own heat bath, we can show that the position and velocity of the monomers, $\mathbf{X}_i$ and $\mathbf{V}_i$, converge (in the sense of distributions) to the solution of (1)–(4) in the limit $\mu \to \infty$.

In reality all beads representing a macromolecule exist within a single heat bath. Thus, we ask whether the correlations introduced by a bath of solvent which interacts with both monomers has a non-negligible affect on the equilibrium statistics of the dimer. Introducing such coupled heat baths for both short-range (in this section) and long-range (in Section IV) interactions we study whether there is a significant difference between the one-bath and two-bath models as we vary ℓ_0, the separation distance, introduced in equation (8). In order to study this problem, we make use of multi-resolution modelling.

A. Multi-resolution model using a co-moving frame

The solvent in the short-range heat bath interacts with the monomers of the dimer through direct contact. In order to simulate the model for long times, i.e. where the dimer has undergone a large excursion, the simulated domain must be vast as will be the number of solvent particles that must be modelled. We present a multi-resolution approach where we only model the solvent that is within the close vicinity of the dimer. We consider a co-moving cubic frame of length L that is centered at $\mathbf{X}_f(t)$, which we here identify with the centre of mass of the dimer at time t, i.e.

$$\mathbf{X}_f(t) = \overline{\mathbf{X}}(t) = \frac{\mathbf{X}_1(t)+\mathbf{X}_2(t)}{2}. \tag{18}$$

Within this frame we explicitly model the heat bath with solvent particles, i.e. they are simulated in the cubic box

$$\mathbf{X}_f(t) + \left[-\frac{L}{2}, \frac{L}{2}\right] \times \left[-\frac{L}{2}, \frac{L}{2}\right] \times \left[-\frac{L}{2}, \frac{L}{2}\right]. \tag{19}$$

Externally we model the heat bath as a continuum, where the particles are distributed according to the spatial Poisson process with density λ_μ given in (15) and the velocities are distributed according to $f_\mu(\mathbf{v})$ given in (16), see Figure 1(a) for a diagrammatic representation of the multi-resolution framework (drawn for clarity in two spatial dimensions, while all our simulations are three-dimensional). As the dimer moves around in $\mathbb{R}^3$ the frame will move with it. In order for the multi-resolution model to capture the full model where solvent particles are distributed in the entire domain, $\mathbb{R}^3$, we need to introduce new solvent particles at the boundary of the frame.

Consider that time is discretized using small time step Δt, i.e. if the current time is t, we want to calculate the state of the system at time $t + \Delta t$. In our simulations of the multi-resolution model we need the probability of introducing a particle at a boundary of frame (19) in a timestep of length Δt and subsequently the distribution of the position $\mathbf{x}_{\text{new}}$ and velocity $\mathbf{v}_{\text{new}}$ of the new solvent particle. For simplicity we transform into the coordinate system of the co-moving frame which over an interval of length Δt has velocity

$$\mathbf{V}_f = \frac{\mathbf{X}_f(t+\Delta t) - \mathbf{X}_f(t)}{\Delta t}. \tag{20}$$

The frame is always translated to occupy the region $[0, L]^3$. Thus, the velocities for the solvent particles in the new reference frame are given by $\mathbf{w}^j = \mathbf{v}^j - \mathbf{V}_f$. We first calculate the density of particles that enter the frame via a particular boundary within a timestep of length Δt. Take, as an illustrative example, the boundary face corresponding to $\{x_1 = 0\}$. Consider particles which are in half-space $(-\infty, 0) \times \mathbb{R}^2$ at time t. These particles have not yet been explicitly included in the simulation. Some of them will be in half-space $(0, \infty) \times \mathbb{R}^2$ at time $t + \Delta t$. Their density, $h(x_1)$, only depends on their first coordinate $x_1 \in (0, \infty)$. We can calculate $h(x_1)$ by integrating density (15)–(16) over solvent particles which are at $x_1' \in (-\infty, 0)$ at time t and have the appropriate velocity to reach $x_1 \in (0, \infty)$ at time $t + \Delta t$, namely as[14]

$$h(x_1) = \int_{-\infty}^0 \int_{\mathbb{R}^2} \lambda_\mu f_\mu\left(\frac{x_1-x_1'}{\Delta t}+V_{f;1}, v_2, v_3\right) dv_2 dv_3 dx_1'$$

$$= \frac{\lambda_\mu}{2}\text{erfc}\left(\frac{x_1+V_{f;1}\Delta t}{\sigma_\mu\Delta t\sqrt{2}}\right), \tag{21}$$

where $V_{f;1}$ is the first component of the frame velocity and $\text{erfc}(z) = 2/\sqrt{\pi}\int_z^\infty \exp(-s^2)\,ds$ is the complementary error function. Integrating (21) over the domain $(0, \infty) \times [0, L] \times [0, L]$ gives us the average number of particles that have entered the frame from the $\{x_1 = 0\}$ boundary in a time interval of length Δt as

$$p_{\text{in}} = \int_0^\infty \int_0^L \int_0^L h(x_1)\,dx_3\,dx_2\,dx_1 \tag{22}$$

$$= \lambda_\mu L^2 \Delta t\left(\frac{\sigma_\mu}{\sqrt{2\pi}}\exp\left[-\frac{V_{f;1}^2}{2\sigma_\mu^2}\right] - \frac{V_{f;1}}{2}\text{erfc}\left[\frac{V_{f;1}}{\sigma_\mu\sqrt{2}}\right]\right).$$
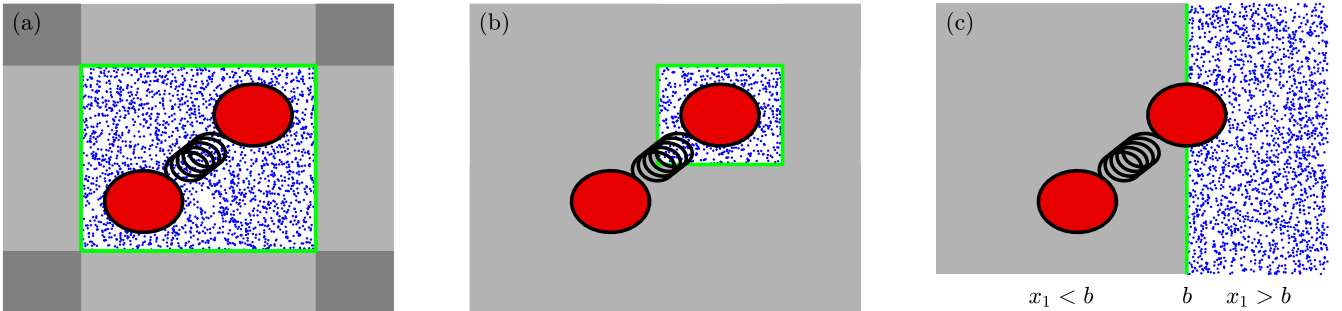
FIG. 1. *A diagrammatic representation of multi-resolution approaches for a dimer in a heat bath with short-range interactions. (a) Simulation of the whole dimer in a co-moving frame. The green box depicts the co-moving frame that is centred about the dimer. The blue dots correspond to solvent molecules that are explicitly modelled. Solvent molecules are not explicitly modelled in the external gray regions. (b) Simulation of one monomer in a co-moving frame. (c) Simulation with a fixed region of space where an MD model is explicitly used. A dimer molecule can move to the gray region where it is simulated using the Langevin description.*

In our simulations we choose a timestep small enough that $p_{in} \ll 1$, we can therefore use p_{in} as the probability of introducing a new solvent particle. Let $\mathbf{z} = [z_1; z_2; z_3]$ be the position of the new solvent particle in the coordinate system of the co-moving frame. Then coordinates z_2 and z_3 are uniformly distributed in $(0, L)$ and the first coordinate can be sampled from the error function distribution

$$C_1 \operatorname{erfc}\left[\frac{z_1 + V_{f;1}\Delta t}{\sigma_\mu \Delta t \sqrt{2}}\right], \qquad \text{for } z_1 \in (0, \infty), \qquad (23)$$

where C_1 is a normalizing constant. Then the position of the new solvent particle in the original coordinates is $\mathbf{x}_{new} = \mathbf{z} + \mathbf{X}_f(t + \Delta t) - [L/2, L/2, L/2]$. The velocity, $\mathbf{w}$, of the new particle in the co-moving frame must have a first coordinate exceeding $z_1/\Delta t$ in order to reach z_1 in a time interval of length Δt. Noting that $\mathbf{w} = \mathbf{v}_{new} - \mathbf{V}_f$ we write down the distribution of the velocity as the following truncated Gaussian distribution

$$C_2 H(v_1 \Delta t - (z_1 + V_{f;1}\Delta t)) f_\mu(\mathbf{v}), \qquad (24)$$

where C_2 is a normalizing constant and $H(\cdot)$ is the Heaviside step function, satisfying $H(y) = 1$ for $y \in [0, \infty)$ and $H(y) = 0$ otherwise. The position and velocity of solvent particles introduced at the other five faces can be done by symmetric modifications of the above distributions.

Random numbers from distributions (23) and (24) can be efficiently sampled using acceptance-rejection algorithms. We use an acceptance-rejection method for the truncated normal distribution (24) presented in the literature[44], while we sample random numbers from the distribution (23) using the acceptance-rejection algorithm presented in Table I. This is a generalization of the acceptance-rejection algorithm for sampling random numbers according to the distribution $\sqrt{\pi} \operatorname{erfc}(z)$ previously used in simulations in the stationary frame[14]. In the case of the distribution (23), we need to sample ran-

- Generate two random numbers η_1 and η_2 uniformly distributed in interval $(0,1)$.
- Calculate $a_1(\beta)$ and $a_2(\beta)$ according to (27) and (28).
- Compute an exponentially distributed random number η_3 by $\eta_3 = -a_1(\beta)\log(\eta_1)$.
- If $\eta_1 \eta_2 < a_2(\beta) \operatorname{erfc}(\eta_3 + \beta)$, then choose η_3 as a sample from the probability distribution (25). Otherwise, repeat the algorithm.

TABLE I. Acceptance-rejection algorithm for sampling random numbers according to the probability distribution $p(z; \beta)$ given by (25).

dom numbers according to the probability distribution

$$p(z; \beta) = C_3(\beta) \operatorname{erfc}(z + \beta), \qquad (25)$$

where $\beta \in \mathbb{R}$ is a constant and $C_3(\beta)$ is the normalizing constant given by

$$C_3(\beta) = \frac{\sqrt{\pi}}{\exp[-\beta^2] - \sqrt{\pi}\,\beta\,\operatorname{erfc}(\beta)}. \qquad (26)$$

The algorithm in Table I does this by generating an exponentially distributed random number η_3 with mean $a_1(\beta)$, where

$$a_1(\beta) = \frac{\sqrt{\pi}}{2} \times \begin{cases} \operatorname{erfc}(\beta)\exp(\beta^2), & \text{for } \beta \geq 0; \\ 1, & \text{for } \beta \leq 0. \end{cases} \qquad (27)$$

To maximise the acceptance probability of this algorithm, we choose its second parameter, $a_2(\beta)$, as

$$a_2(\beta) = \begin{cases} 1/\operatorname{erfc}(\beta), & \text{for } \beta \geq 0; \\ \exp\left(2\beta/\sqrt{\pi}\right), & \text{for } \beta \leq 0. \end{cases} \qquad (28)$$

Then its acceptance probability is depending on β as

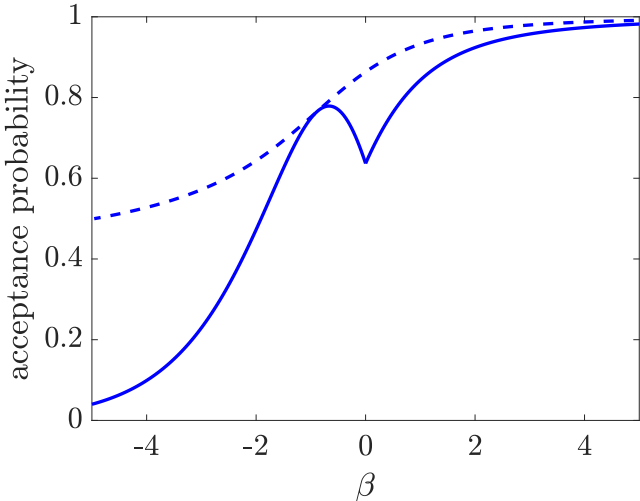$$\frac{a_2(\beta)}{a_1(\beta)\, C_3(\beta)}. \qquad (29)$$

FIG. 2. *Plot of the acceptance probability* (29) *of the algorithm presented in Table* I *for parameters given by* (27) *and* (28) *(solid line) compared with the acceptance probability* (29) *calculated for optimal choices of* $a_1(\beta)$ *and* $a_2(\beta)$ *for each parameter value* β.

We plot the acceptance probability (29) in Figure 2 for our choices (27)-(28) of $a_1(\beta)$ and $a_2(\beta)$ as the solid line.

We observe that the acceptance probability (29) for $\beta = 0$ is equal to $2/\pi \approx 63.7\%$. This value can be improved[14] in the case of $\beta = 0$ to 86.3% provided that we choose $a_1 = 0.532$ and $a_2 = 0.814$. To obtain a similar improvement for all values of β, we could choose both $a_1(\beta)$ and $a_2(\beta)$ to maximise the acceptance probability (29), rather than postulating that $a_1(\beta)$ is given by the piecewise defined function (27) and optimizing $a_2(\beta)$ only. The acceptance probability (29) of the resulting algorithm (which would have $a_1(\beta)$ and $a_2(\beta)$ given by a lookup table, rather than by using formulas (27)-(28)) is plotted in Figure 2 as the dashed line for comparison. However, in our illustrative simulations, we use the acceptance-rejection algorithm in Table I with the values of $a_1(\beta)$ and $a_2(\beta)$ given by (27)-(28).

Comparing equations (25) and (23), we observe that we can sample random numbers from the distribution (23) by sampling random numbers from the distribution $p(z; V_{f;1}\Delta t)$ (using the acceptance-rejection algorithm in Table I for $\beta = V_{f;1}\Delta t$)) and multiplying them by the factor $\sigma_\mu \Delta t \sqrt{2}$.

One iteration (i.e. an update of the state of the system from time t to time $t + \Delta t$) of the multi-resolution simulation algorithm in the co-moving frame is given as Algorithm [S1]–[S7] in Table II. It evolves the positions and velocities of both monomers together with the positions and velocities of $N(t)$ solvent particles, where $N(t)$ does depend on time t. To formulate Algorithm [S1]–[S7], we assume that the timestep Δt is chosen small enough so that at most one collision happens per iteration.

We initialize the two monomers with a separation dis-

tance ℓ_0 and generate a Poisson number (with mean $\lambda_\mu L^3$) of solvent particles in our simulation domain, the cubic frame (19). The solvent particles are initially placed uniformly in the frame (19), where we remove particles overlapping with monomers (before we begin our simulation) to get the initial number, $N(0)$, of simulated solvent particles. Their initial velocities are drawn from the Maxwell-Boltzmann distribution (16).

In Step [S1], we update the system over the time interval $(t, t + \Delta t]$ using the "free-flight" positions for each monomer and solvent particle, namely we use

$$\widehat{\mathbf{X}}_i(t + \Delta t) = \mathbf{X}_i(t) + \mathbf{V}_i(t)\,\Delta t, \tag{30}$$

$$\widehat{\mathbf{x}}_i^j(t + \Delta t) = \mathbf{x}_i^j(t) + \mathbf{v}_i^j(t)\,\Delta t, \tag{31}$$

where $i = 1, 2$ is the monomer number and $j = 1, 2, \ldots, N(t)$, is the number of the heat bath particle. Since Δt is chosen so small that only one collision happens during the time interval $[t, t + \Delta t)$, most of the "free-flight" positions of solvent particles are accepted in Step [S2] as their updated positions $\mathbf{x}_i^j(t + \Delta t)$ and only the solvent particle colliding with a monomer is further updated.

In Step [S3], we update the velocities of the monomers by solving (10)–(11) over one time step $[t, t + \Delta t]$. We discretize (10)–(11) using the forward Euler method as follows

$$\mathbf{V}_1(t + \Delta t) = \widetilde{\mathbf{V}}_1 + \frac{\Phi'(R)}{M}\frac{\mathbf{R}}{R}\,\Delta t, \tag{32}$$

$$\mathbf{V}_2(t + \Delta t) = \widetilde{\mathbf{V}}_2 - \frac{\Phi'(R)}{M}\frac{\mathbf{R}}{R}\,\Delta t, \tag{33}$$

where $\widetilde{\mathbf{V}}_i$, for $i = 1, 2$, is either the post collision velocity (if a collision happened in Step [S2]) or is equal to $\mathbf{V}_i(t)$. In Steps [S4]–[S5], we update the position and velocity of the frame. We remove solvent particles which are outside of the simulation domain and update $N(t)$ accordingly.

In Step [S6], we use probability p_{in}, given by (22), to check whether any solvent particle entered the simulation domain during the time interval $(t, t + \Delta t]$. Since p_{in} is the probability of entering the domain through one of its six sides, we can, for time step Δt small enough that $6p_{in} \ll 1$, introduce at most one solvent particle through a randomly chosen side with probability $6p_{in}$. The initial position and velocity of the introduced solvent particle are sampled according to distributions (23) and (24) or their symmetric modifications, taking into account through which side of the cubic frame (19) the particle entered the frame.

There is one little caveat in our derivation of p_{in}. To derive equation (22) we integrated over the half-space $(-\infty, 0) \times \mathbb{R}^2$, meaning that once we consider all six faces of the cubic frame (19) we have over-counted twice at the edges and three times at the corners (as it is highlighted with darker gray shading in our illustrative diagram in Figure 1(a)). This will have negligible effect if we choose L sufficiently large. However, it can bias our simulation for values of L comparable with the monomer size

[S1]	Update the positions of the solvent and the monomers by their "free-flight" positions (30)–(31).
[S2]	If the "free-flight" position (31) of a solvent particle lies within the radius of either of the monomers, reverse the trajectories of the solvent and the monomer by time $\tau < \Delta t$ such that they are just touching. Calculate post-collision velocities by equations (13)–(14) and update their new positions by moving forward by time τ. Otherwise, each "free-flight" position is accepted as the particle's position at time $t + \Delta t$.
[S3]	Update the velocities of the monomers by (32)–(33).
[S4]	Calculate the new centre of the co-moving frame, $\mathbf{X}_{\mathrm{f}}(t + \Delta t)$, by (18). Update $N(t)$ by removing solvent particles which now lie outside of the frame (19) from the simulation.
[S5]	Calculate the velocity of the frame, $\mathbf{V}_{\mathrm{f}}$, over the interval $[t, t + \Delta t]$ by equation (20).
[S6]	Generate two random number r_1 and r_2 uniformly distributed in interval $(0, 1)$. If $r < 6p_{\mathrm{in}}$, then choose a side of the cube at random and generate proposed position $\mathbf{x}_{\mathrm{new}}$ and velocity $\mathbf{v}_{\mathrm{new}}$ of the new solvent particle according to distributions (23) and (24). If $r_2 < h_{\mathrm{acc}}(\mathbf{x}_{\mathrm{new}}, \mathbf{v}_{\mathrm{new}})$, then increase $N(t) = 1$ and initialize the new solvent particle at position $\mathbf{x}_{\mathrm{new}}$ with velocity $\mathbf{v}_{\mathrm{new}}$.
[S7]	Continue with step [S1] using time $t = t + \Delta t$.

TABLE II. One iteration of the multi-resolution simulation algorithm of the dimer in a co-moving frame.

r_0 when Δt is not sufficiently small as boundary effects become more pronounced. To compensate for this effect, we consider the sampled position, $\mathbf{x}_{\mathrm{new}}$ and velocity $\mathbf{v}_{\mathrm{new}}$ of the new incoming particle at time $t + \Delta t$ and calculate its previous position at time t by

$$\mathbf{y} = \mathbf{x}_{\mathrm{new}} - \mathbf{v}_{\mathrm{new}} \Delta t.$$

If $\mathbf{y}$ is in the regions which were counted twice or three times in our derivation, we reject the proposed introduction of the new solvent particle with the corresponding probability. Namely, we use the acceptance probability in Step [S6] given by

$$h_{\mathrm{acc}}(\mathbf{x}_{\mathrm{new}}, \mathbf{v}_{\mathrm{new}}) = \begin{cases} 1, & \text{for } \mathbf{y} - \mathbf{X}_{\mathrm{f}}(t) \in \mathcal{Y}_1; \\ 1/2, & \text{for } \mathbf{y} - \mathbf{X}_{\mathrm{f}}(t) \in \mathcal{Y}_2; \\ 1/3, & \text{for } \mathbf{y} - \mathbf{X}_{\mathrm{f}}(t) \in \mathcal{Y}_3, \end{cases}$$

where $\mathcal{Y}_j \subset \mathbb{R}^3$ is the region of the space which consists of points which have exactly j of their coordinates outside of the interval $[-L/2, L/2]$. For example, in our two-dimensional diagrammatic representation in Figure 1(a), the lighter gray shading corresponds to region $\mathcal{Y}_1$ while the darker gray shading corresponds to region $\mathcal{Y}_2$.

In our illustrative simulations, we use algorithm [S1]–[S7] from Table II together with parameter values $r_0 = 0.08$, $\gamma = 10$, $D = 1$, $\mu = 10^3$, $k = 10^6$ and $L = 0.72$ for the one-bath case. In Figure 3, we compare simulation
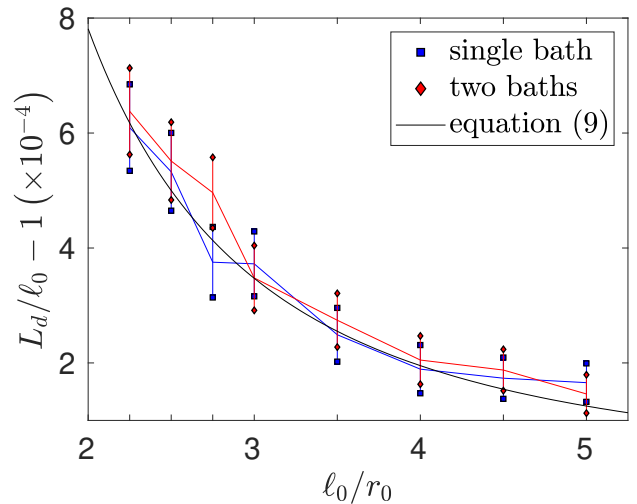

FIG. 3. *The extension of the average length of a dimer from its separation distance ℓ_0. The equilibrium data for each model was collected from a long-time simulation of length 100 dimensionless time units where $\Delta t = 10^{-6}$ and the monomers were initially placed with separation ℓ_0. The values of $\alpha = \ell_0/r_0$ presented are $\{2.25, 2.5, 2.75, 3, 3.5, 4, 4.5, 5\}$. The parameters used are $r_0 = 0.08$, $\gamma = 10$, $D = 1$, $\mu = 10^3$, and $k = 10^6$. In the one-bath case we use $L = 0.72$ for the frame (19) enclosing the whole dimer, and for the two-bath case, we use $L = 0.32$ for each monomer frame.*

results of the average length of the dimer at equilibrium, L_d, for the one-bath and two-bath models. Since the two-bath case uses uncoupled heat baths, we can further improve the efficiency of our algorithm by centering the co-moving frame corresponding to each heat bath on the corresponding monomer, i.e. we use $\mathbf{X}_{\mathrm{f}}(t) = \mathbf{X}_i(t)$ for the heat bath corresponding to the i-th monomer in Step [S4] (instead of the centre of mass (18)) and choose smaller value of L in the two-bath case, namely $L = 0.32$. In both one-bath and two-bath models, the solvent particles are distributed according to the spatial Poisson process with density λ_μ given by (15). The velocities are distributed according to the Maxwell-Boltzmann distribution $f_\mu(\mathbf{v})$ given by (16). We note that in the two-bath case, our model converges to the Langevin dynamics (1)–(4) as $\mu \to \infty$. This allows us to attribute any changes between the one-bath case and the Langevin model to the correlations induced by sharing a heat bath. The asymptotic analytic result obtained for the Langevin model, equation (9), is plotted as the black solid line for comparison.

In Figure 3, we set the separation distance to be $\ell_0 = \alpha r_0$ where $\alpha \geq 2$, such that at this distance apart the monomers are not overlapping. The plot shows the two-sided 99% confidence intervals for $(L_d - \ell_0)/\ell_0$ for $\alpha \in \{2.25, 2.5, 2.75, 3, 3.5, 4, 4.5, 5\}$. Firstly, we note that $L_d > \ell_0$ in each of the models as predicted in (9). There seems to be reasonable correspondence between the one- and two-bath models, with the confidence intervals over-

lapping. This suggests that the correlations we lose by approximating a larger co-moving frame around both monomers with two smaller dedicated frames around each monomer are negligible, allowing us to increase efficiency without biasing our overall results. In the next section, we build on this observation and present a multi-resolution framework which replaces one of the smaller dedicated frames by a coarser model of the heat bath, written in terms of the Langevin dynamics.

B. Monomers with different resolution

As the length of a polymer (i.e. numbers of monomers) increases, a model incorporating solvent particles around each of the monomers becomes increasingly computationally expensive. However, a fully coarse-grained Langevin model of a polymer such as the Rouse model[39] can lack the required level of detail. Thus, some multi-resolution approaches for simulating macromolecules only model an important (small) part of a macromolecule using a detailed modelling approach[38–42]. In our case, we can mimic such methodologies by modelling the first monomer with explicit solvent with a heat bath of physical molecules, while the second monomer is modelled using the Langevin equations (3) and (4). Such a multi-resolution approach is schematically shown in Figure 1(b). To simulate this model we use a co-moving frame, given by equation (19), which is centered around the first monomer, i.e. $\mathbf{X}_f(t) = \mathbf{X}_1(t)$.

One iteration of the algorithm is presented as Algorithm [M1]–[M5] in Table III. To begin, we initialize the particle positions and velocities in the similar way as in the case of Algorithm [S1]–[S7], with the only difference that the cubic frame (19) is now centered around the first monomer. Steps [M1] and [M2] are directly equivalent to steps [S1] and [S2]. In Step [M3], we update the position and velocity of the second monomer by

$$\mathbf{V}_2(t + \Delta t) = \mathbf{V}_2(t) - \left(\frac{\Phi'(R)}{M} \frac{\mathbf{R}}{R} + \gamma \mathbf{V}_2(t) \right) \Delta t + \gamma \sqrt{2D\Delta t}\, \xi, \quad (34)$$

where ξ is sampled from the normal distribution with zero mean and unit variance. That is, we have replaced the heat bath of the second monomer by solving the corresponding Langevin equation (1)–(4) using the standard Euler-Maruyama integrator. There have been other schemes developed in the literature for discretizing the Langevin equation such as van Gunsteren and Berendsen[45] and the Langevin Impulse integrators, which capture the Langevin dynamics more accurately especially in the presence of forces, such as the spring force between the monomers[46]. Another option would be to consider the BBK integrator[47], which we use in Section IV A, where we present a multi-resolution algorithm for the long-range interaction heat bath model and discretize the Langevin equation using a combination of

[M1]	Update the positions of the solvent and the monomers by their "free-flight" positions (30)–(31).
[M2]	If the "free-flight" position (31) of a solvent particle lies within the radius of the first monomer, reverse the trajectories of the solvent and the monomer by time $\tau < \Delta t$ such that they are just touching. Calculate post-collision velocities by equations (13)–(14) for $i = 1$ and update their new positions by moving forward by time τ. Otherwise, each "free-flight" position is accepted as the particle's position at time $t + \Delta t$.
[M3]	Update the velocity of the first monomer by (32) and the velocity of the second monomer by (34).
[M4]	Calculate the new centre of the co-moving frame as $\mathbf{X}_f(t + \Delta t) = \mathbf{X}_1(t)$. Update $N(t)$ by removing solvent particles which now lie outside of the frame (19) from the simulation. Use steps [S5]–[S6] from the algorithm in Table II to introduce new solvent particles into the co-moving frame (19).
[M5]	Continue with step [M1] using time $t = t + \Delta t$.

TABLE III. One iteration of the multi-resolution simulation algorithm of the dimer in the heat bath with short-range interactions, where the second monomer is simulated by the Langevin dynamics.

the velocity Verlet and Euler-Maruyama integrators, see equations (45)-(49). An additional approach is the Verlet scheme[48] that approximates the velocity using a central difference discretization rather than the forward difference approach used in the Euler-Maruyama method, or Runge-Kutta methods[49], which could further reduce the error of the multi-resolution simulations.

In order to compare simulations of the multi-resolution model with simulations of the Langevin model (1)–(4) we use the velocity autocorrelation function of the dimer, $C_d(\tau)$, given by equation (5). It has been analytically calculated for the Langevin description in equation (6). In Figure 4, we present numerical estimates of the velocity autocorrelation function of the multi-resolution model from long time simulation data, using definition (5).

Our results compare well with the theoretical result for the Langevin model, though it seems like there is a slightly raised value for $C_d(0)$. Using (7), we can estimate the diffusion constant of the dimer D_d by numerically integrating the velocity auto-correlation function in interval $[0, 1]$. We obtain $D_d \approx 0.529$, while its theoretical value for the dimer model is given in equation (7) as $D/2 = 0.5$. Another approach is to fit the exponential function, in the form equation (6), to the computational result presented in Figure 4. In this way, the values of both D and γ can be estimated simultaneously. We found that $D \approx 1.0714$, which is higher than our parameter value $D = 1$, and $\gamma \approx 9.6064$, which is lower than $\gamma = 10$ used in our simulations. This could suggest that the value of λ_μ is too low or that of σ_μ is too high in our simulations. However, when these quantities are measured during the simulations we do not observe any deviation.
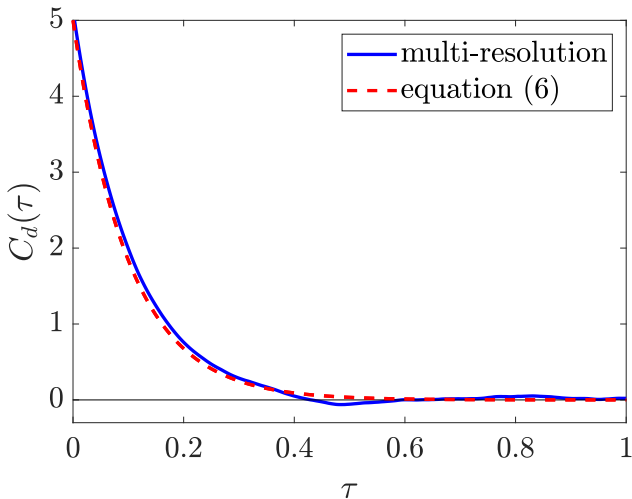
FIG. 4. *The velocity autocorrelation function for the multi-resolution model (blue solid line) for short-range interactions. The function is estimated from long time simulation over dimensionless time of 500 time units. It is compared with the result for the Langevin description of the whole dimer, given by equation (6) (red dashed line). The parameters are $r_0 = 0.08$, $\gamma = 10$, $D = 1$, $\mu = 10^3$, $k = 10^6$, $\ell_0 = 4r_0$, $\Delta t = 10^{-6}$ and $L = 0.32$.*

This suggests that, rather than our sampling methods, there are small errors introduced by our implementation of the moving frame, or more profound boundary effects introduced by the small size of the frame. A potential problem in the implementation of the co-moving frame, is that solvent particles that leave the frame never return. For a stationary frame this is valid as the monomer cannot interact with a particle that leaves. However, for a co-moving small frame centred about the monomer, a solvent particle could leave the frame and return at a later time in the simulation. This is not taken into account in the presented algorithms.

IV. LONG-RANGE INTERACTION HEAT BATH

Coarse-grained models of molecular systems can be written in terms of beads interacting through coarse-grained force fields. Each bead represents a collection of atoms and a coarse-grained potential energy can be constructed from detailed all-atom MD. Such an approach can usually provide a good description of equilibrium properties of molecular systems, but it does not necessarily lead to correct dynamics if the time evolution of the system is solely based on the Hamiltonian dynamics corresponding to the coarse-grained potential energy surface[50]. Dynamical behaviour can be corrected by introducing additional degrees for freedom (fictitious particles) interacting with each coarse-grained bead[50–52]. Fictitious particles can then be subject to suitable friction and noise terms to correct the dynamics.

Considering our dimer molecule model as an example of a coarse-grained molecule, written in terms of two coarse-grained beads (monomers) interacting through coarse-grained potential energy (8), then each monomer could be coupled with one or several fictitious particles interacting with the monomer through a suitable harmonic spring term[50,51]. Our long-range interaction heat bath is based on this approach, by assuming that the i-th monomer, $i = 1, 2$, is coupled with N_i harmonic oscillators, in a manner similar to well known theoretical heat bath models[15,16]. Then equations (10)–(11), expressing Newton's second law of motion, include additional terms as follows[15]

$$M\frac{\mathrm{d}\mathbf{V}_1}{\mathrm{d}t} = \Phi'(R)\frac{\mathbf{R}}{R} + \sum_{j=1}^{N_1} k_{1,j}\,\alpha_{1,j}\Big(\mathbf{x}_1^j - \alpha_{1,j}\mathbf{X}_1\Big), \quad (35)$$

$$M\frac{\mathrm{d}\mathbf{V}_2}{\mathrm{d}t} = -\Phi'(R)\frac{\mathbf{R}}{R} + \sum_{j=1}^{N_2} k_{2,j}\,\alpha_{2,j}\Big(\mathbf{x}_2^j - \alpha_{2,j}\mathbf{X}_2\Big), \quad (36)$$

where $\mathbf{x}_i^j$ is the position of the j-th solvent particle which interacts with the i-th monomer through a harmonic spring with spring constant $k_{i,j}$ and interaction constants $\alpha_{i,j}$, $j = 1, 2, \ldots, N_i$, $i = 1, 2$. Equations (35)–(36) are coupled with the evolution equations for solvent particles. We assume that $\mathbf{v}_i^j$ is the velocity of the j-th solvent particle interacting with the i-th monomer. Moreover, we assume that all oscillators have the same mass, m. Using Newton's second law of motion, we get the following evolution equations for the heat bath oscillators

$$\frac{\mathrm{d}\mathbf{x}_i^j}{\mathrm{d}t} = \mathbf{v}_i^j, \qquad (37)$$

$$m\frac{\mathrm{d}\mathbf{v}_i^j}{\mathrm{d}t} = -k_{i,j}\Big(\mathbf{x}_i^j - \alpha_{i,j}\mathbf{X}_i\Big), \qquad (38)$$

for $j = 1, 2, \ldots, N_i$ and $i = 1, 2$. Unlike in some fictitious particle models[50–52], we do not include friction and random forces into equation (38) for solvent, because we assume that we explicitly model all solvent particles, i.e. N_1 and N_2 are considered to satisfy $N_1 \gg 1$ and $N_2 \gg 1$. We are therefore working 'close' to the limit $N_1 \to \infty$ and $N_2 \to \infty$, in which we can get the convergence of our long-range interaction heat bath to the Langevin dynamics as discussed below. In practice, it is impossible to include all solvent molecules in simulations and friction and noise terms are still included to control temperature of the simulated system[2,53]. We can solve the solvent equations of motion (37)–(38) to give[2,54]

$$\mathbf{x}_i^j = \mathbf{x}_i^j(0)\cos(\omega_{i,j}t) + \frac{\mathbf{v}_i^j(0)}{\omega_{i,j}}\sin(\omega_{i,j}t)$$
$$+ \alpha_{i,j}\,\omega_{i,j}\int_0^t \sin(\omega_{i,j}(t-\tau))\,\mathbf{X}_i(\tau)\,\mathrm{d}\tau$$

where $\mathbf{x}_i^j(0)$ is the initial position of the j-th heat bath particle corresponding to the i-th monomer, $\mathbf{v}_i^j(0)$ is its initial velocity and $\omega_{i,j} = (k_{i,j}/m)^{1/2}$ is its frequency.

Substituting for $\mathbf{x}_1^j$ and $\mathbf{x}_2^j$ in dimer's equations of motion (35)–(36), we obtain the following coupled system of generalized Langevin equations

$$M\frac{d\mathbf{V}_1}{dt} = \Phi'(R)\frac{\mathbf{R}}{R} - \int_0^t \kappa_1(\tau)\,\mathbf{V}_1(t-\tau)\,d\tau + \boldsymbol{\xi}_1, \quad (39)$$

$$M\frac{d\mathbf{V}_2}{dt} = -\Phi'(R)\frac{\mathbf{R}}{R} - \int_0^t \kappa_2(\tau)\,\mathbf{V}_2(t-\tau)\,d\tau + \boldsymbol{\xi}_2, \quad (40)$$

where the friction kernel $\kappa_i(\tau)$ and noise term $\boldsymbol{\xi}_i \equiv \boldsymbol{\xi}_i(t) = [\xi_{i;1}, \xi_{i;2}, \xi_{i;3}]$ are given by

$$\kappa_i(\tau) = m\sum_{j=1}^{N_i} \alpha_{i,j}^2\,\omega_{i,j}^2\cos(\omega_{i,j}\tau),$$

$$\boldsymbol{\xi}_i(t) = m\sum_{j=1}^{N_i} \mathbf{x}_i^j(0)\,\alpha_{i,j}\,\omega_{i,j}^2\cos(\omega_{i,j}t)$$
$$+ \mathbf{v}_i^j(0)\,\alpha_{i,j}\,\omega_{i,j}\sin(\omega_{i,j}t),$$

for $i = 1, 2$. We assume that initial positions and velocities of solvent oscillators, $\mathbf{x}_i^j(0)$ and $\mathbf{v}_i^j(0)$, are both independently sampled according to their equilibrium distributions. Then noise autocorrelation function is given by the generalized fluctuation-dissipation theorem

$$\lim_{t\to\infty}\langle\xi_{i;j}(t)\,\xi_{i;n}(t-\tau)\rangle = 2k_BT\,\delta_{j,n}\,\kappa_i(\tau),$$

where k_B is the Boltzmann constant and T is the absolute temperature. Next, we assume that the frequencies $\omega_{i,j}$ are sampled from a (continuous) exponential distribution with mean $\overline{\omega}$ and we set our interaction constants equal to

$$\alpha_{i,j} = \frac{1}{\omega_{i,j}}\sqrt{\frac{2\gamma\overline{\omega}}{N_i\,m\,\pi}}, \quad (41)$$

where $\gamma > 0$ is the friction constant used in equations (2) and (4). Then friction kernel (41) becomes

$$\kappa_i(\tau) = \frac{2\gamma\overline{\omega}}{\pi}\frac{1}{N_i}\sum_{j=1}^{N_i}\cos(\omega_{i,j}\tau).$$

Passing to the limit $N_i \to \infty$ allows us to consider the above summation as a continuous integral over the distribution of oscillator frequencies, with both friction kernels $\kappa_1(\tau)$ and $\kappa_2(\tau)$ converging to the same friction kernel[54]

$$\kappa(\tau) = \frac{2\gamma}{\pi}\int_0^\infty\cos(\omega\tau)\exp\left(-\frac{\omega}{\overline{\omega}}\right)d\omega$$
$$= \frac{2\gamma}{\pi}\frac{\overline{\omega}}{\overline{\omega}^2\tau^2+1}. \quad (42)$$

Then $\int_0^\infty\kappa(\tau)\,d\tau = \gamma$. Moreover, we can define the limiting friction kernel by

$$\kappa_\infty(\tau) = \lim_{\overline{\omega}\to\infty}\kappa(\tau),$$

which, for our choice of oscillators' frequencies and interaction terms (41), satisfies $\kappa_\infty(\tau) = 0$ for $\tau > 0$ and
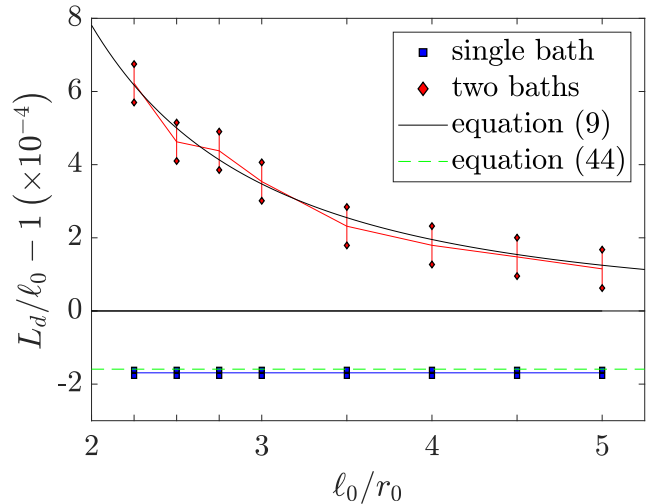

FIG. 5. *The extension of the average length of a dimer from its separation distance ℓ_0 for long-range interaction heat bath models. The values of parameters are the same as in Figure 3, together with $\overline{\omega} = 100$, $N_1 = N_2 = N = 10^4$, $M = 1$ and $m = 10^{-3}$, which give the same value of $\mu = M/m$ as used in Figure 3. The simulations for the single heat bath case use parameter choice (43) with $\alpha_j^2 = \gamma\overline{\omega}/(N\pi k_j)$, $k_j = m\omega_j^2/2$, and ω_j sampled according to the exponential distribution with mean $\overline{\omega}$, confirming result given in equation (44) (green dashed line). The results for the two heat bath case are compared with the result obtained for the Langevin model in equation (9) (black solid line).*

$\kappa_\infty(0) = \infty$. Thus the limiting kernel is a multiple of the Dirac delta function. Therefore the position and velocity of the monomers, $\mathbf{X}_i$ and $\mathbf{V}_i$, converge to the solution of (1)–(4) in the limit $\overline{\omega} \to \infty$, provided that each monomer has its own separate heat bath. Moreover, we obtain the Einstein-Smoluchowski relation for the diffusion constant of the monomer as $D = k_BT/(\gamma M)$.

As in Section III, we have explained our MD model of the dimer using the case where each monomer has its own heat bath. We now turn our attention to the case when monomers share their heat bath. This has been studied in the case of the short-range interaction MD model in Section III A with the help of multi-resolution modelling in a co-moving frame, as schematically shown in Figure 1(a). In the case of long-range interactions, a co-moving frame is less straightforward to implement because we need to take into account that particles outside of the simulated box do exert (long-range) forces on particles in our simulation domain. Some multi-resolution techniques in the literature solve this problem by introducing suitable overlap (bridging, blending) regions[51,55-57], where molecules which are near the simulation domain exert some partial forces on the simulated molecules.

In what follows, we do not truncate the simulated domain, but we consider a different multi-resolution approach in Section IV A. Before then we discuss results comparable to Figure 3, i.e. we compare simulations with

a single heat bath and two heat baths for the case of our long-range interaction MD model. The results are presented in Figure 5, where we use the same values of ℓ_0 as in Figure 3, expressed as α-multiples of r_0, although our long-range interaction model does not make use of parameter r_0. The value of L_d is for each value of ℓ_0 calculated from a long simulation over 200 dimensionless time units, where the first 100 time units are used to equilibrate the system, while the second half of each simulation is used to compute L_d. To initialise this model we start with monomers separated by the rest length ℓ_0 and sample oscillators' frequencies, $\omega_{i,j}$, according to the exponential distribution with mean $\overline{\omega}$. Their positions and velocities are sampled from the Maxwell-Boltzmann distribution. For the two-bath model, each dimer particle is separately initialised with its own set of oscillators around their respective positions in space.

In Figure 5, we observe that in the case of the two-bath model we obtain results which match well with equation (9) for our parameter values. These results are also directly comparable with the results obtained for the two-bath case in Figure 3. The situation is more complicated in the case of simulations with a single heat bath with N oscillators. Then, using notation (12), we can rewrite (35)–(36) as

$$M\frac{\mathrm{d}\mathbf{V}_i}{\mathrm{d}t} = (-1)^{i+1}\Phi'(R)\,\frac{\mathbf{R}}{R} + \sum_{j=1}^{N} k_{i,j}\,\alpha_{i,j}\Big(\mathbf{x}^j - \alpha_{i,j}\mathbf{X}_i\Big),$$

for $i = 1, 2$, where the heat bath evolution equation (38) includes terms corresponding to both monomers

$$m\frac{\mathrm{d}\mathbf{v}^j}{\mathrm{d}t} = -k_{1,j}\left(\mathbf{x}^j - \alpha_{1,j}\mathbf{X}_1\right) - k_{2,j}\left(\mathbf{x}^j - \alpha_{2,j}\mathbf{X}_2\right),$$

for $j = 1, 2, \ldots, N$. Our results will then depend how we choose parameters $k_{i,j}$ and $\alpha_{i,j}$. For example, if we choose $k_{i,j}$ and $\alpha_{i,j}$ to be the same for both monomeres, i.e.

$$k_{1,j} = k_{2,j} = k_j, \quad \text{and} \quad \alpha_{1,j} = \alpha_{2,j} = \alpha_j, \qquad (43)$$

for $j = 1, 2, \ldots, N$, then the oscillating frequency of the j-th heat bath oscillator is $\omega_j = \sqrt{2k_j/m}$ and we can subtract the evolution equations for monomers to obtain

$$M\frac{\mathrm{d}^2\mathbf{R}}{\mathrm{d}t^2} = -2\Phi'(R)\,\frac{\mathbf{R}}{R} - \sum_{j=1}^{N} k_j\,\alpha_j^2\,\mathbf{R}.$$

This equation does not contain any heat bath variables. Using (41) to select α_j, i.e. using $\alpha_j^2 = 2\gamma\overline{\omega}/(N\,m\,\pi\,\omega_j^2) = \gamma\overline{\omega}/(N\,\pi\,k_j)$, we get

$$M\frac{\mathrm{d}^2\mathbf{R}}{\mathrm{d}t^2} = -2\Phi'(R)\,\frac{\mathbf{R}}{R} + \frac{\gamma\overline{\omega}}{\pi}\,\mathbf{R}.$$

Using potential (8), we conclude that we effectively obtain a shorter rest length of the spring which gives the following approximation

$$L_d \approx \frac{2\,k\,\pi\,\ell_0}{2\,k\,\pi + \gamma\overline{\omega}}. \qquad (44)$$

[L1] Update velocities of the dimer and solvent particles for a half time step using (45).

[L2] Update positions of the dimer and solvent particles using (46).

[L3] Recalculate accelerations of each monomer and solvent oscillators by (48), (49) and (50).

[L4] Update velocities of the dimer and solvent particles for a half time step using (47).

[L5] Continue with step [L1] using time $t = t + \Delta t$.

TABLE IV. One iteration of the multi-resolution simulation algorithm of the dimer in the heat bath with long-range interactions, where the second monomer is simulated by the Langevin dynamics.

This result is plotted in Figure 5 together with results obtained by illustrative simulations. We use a long-time simulation of length 200 dimensionless time units, with monomers initially placed at separation ℓ_0 and averaging over the second half of the simulation (of length 100 dimensionless time units) to obtain the presented values of dimer's expected length L_d.

In Figure 5, we observe that the average dimer length, L_d, during our single heat bath simulations is smaller than the natural length of the spring, ℓ_0. However, this conclusion is only a consequence of our choice of parameters (43). An opposite phenomenon can be observed in simulations for other parameter regimes. For example, if we divide our oscillators into two groups consisting of N_1 and N_2 oscillators, i.e. $N = N_1 + N_2$, and choose our parameters $k_{i,j}$ and $\alpha_{i,j}$ such that

$$k_{2,j} = 0, \qquad \text{for } j = 1, 2, \ldots, N_1,$$
$$k_{1,j} = 0, \qquad \text{for } j = N_1 + 1, N_1 + 2, \ldots, N,$$

then our "one-bath" case is effectively equal to the two-bath case for which we have the result given in equation (9) presented in Figure 5. In particular, depending on our choices of $k_{i,j}$ and $\alpha_{i,j}$, the single heat bath case can both increase or decrease the average length of the dimer.

A. Multi-resolution modelling of dimer

In Figure 1(b), we use our dimer example to illustrate a multi-resolution approach which models a part of a molecule using a detailed MD approach, while using a coarser description of the rest of the molecule. Here, in the same manner as carried out for our short-range model in Section III B, we illustrate such a multi-resolution approach using our long-range interaction MD model. We use the Langevin model (1)–(4) to coarse-grain one of the monomers, while the other monomer is modelled in detail using the MD model with its heat bath described by harmonic oscillators (37)–(38). As in Figure 4, we again calculate numerical estimates for the velocity autocorrelation function, $C_d(\tau)$ in equation (5), from long

time simulations of the dimer after equilibrium has been reached.

The pseudo-code of one iteration our multi-resolution algorithm is presented as Algorithm [L1]–[L5] in Table IV. Algorithm [L1]–[L5] is based on the velocity Verlet integrator, where both monomers are updated by

$$\mathbf{V}_i\left(t + \tfrac{1}{2}\Delta t\right) = \mathbf{V}_i(t) + \frac{1}{2}\,\mathbf{A}_i(t)\,\Delta t, \qquad (45)$$

$$\mathbf{X}_i(t + \Delta t) = \mathbf{X}_i(t) + \mathbf{V}_i(t + \tfrac{1}{2}\Delta t)\,\Delta t, \qquad (46)$$

$$\mathbf{V}_i(t + \Delta t) = \mathbf{V}_i\left(t + \tfrac{1}{2}\Delta t\right) + \frac{1}{2}\,\mathbf{A}_i(t + \Delta t)\,\Delta t, \quad (47)$$

where $\mathbf{A}_i$, for $i = 1, 2$, is the acceleration of the corresponding monomer. For the first monomer, its acceleration $\mathbf{A}_1$ is defined as the right hand side of equation (35) divided by M, i.e.

$$\mathbf{A}_1 = \frac{\Phi'(R)}{M}\,\frac{\mathbf{R}}{R} + \frac{1}{M}\sum_{j=1}^{N_1} k_{1,j}\,\alpha_{1,j}\left(\mathbf{x}_1^j - \alpha_{1,j}\mathbf{X}_1\right). \quad (48)$$

For the second monomer, we use the BBK integrator[47], i.e. we define its acceleration as

$$\mathbf{A}_2 = -\frac{\Phi'(R)}{M}\,\frac{\mathbf{R}}{R} - \gamma\mathbf{V}_2 + \gamma\,\sqrt{\frac{2D}{\Delta t}}\,\xi, \qquad (49)$$

where ξ is sampled from the normal distribution with zero mean and unit variance. The corresponding solvent oscillator integrator is identical to the scheme (45)–(47), with $\mathbf{X}_1$, $\mathbf{V}_1$ and $\mathbf{A}_1$ replaced by $\mathbf{x}^j$, $\mathbf{v}^j$ and $\mathbf{a}^j$, respectively, where acceleration $\mathbf{a}^j$ is defined as the right hand side of equation (38) divided by m, i.e.

$$\mathbf{a}^j = -\frac{k_{i,j}}{m}\left(\mathbf{x}_i^j - \alpha_{i,j}\mathbf{X}_i\right). \qquad (50)$$

The results obtained by Algorithm [L1]–[L5] are compared with analytic results given by equation (6) for the Langevin model in Figure 6. We see that there is a good correspondence between these, suggesting that the value $\bar{\omega} = 100$ is large enough to create an accurate Dirac delta approximation from the kernel function (42), along with having a large enough number of oscillators, $N_1 = 10^5$, in our heat bath for our other approximations to hold. If these conditions did not hold, we would see that our kernel function has a different form (for example, decaying at a slower rate), and in this case we would have to use a generalized Langevin model as our coarse-graining approach in order to capture the dynamics of the dimer with sufficient accuracy.

The diffusion constant of the dimer, D_d, can again be estimated by numerically integrating the velocity autocorrelation function. Integrating our results from Figure 6 over interval $[0,1]$, we obtain $D_d \approx 0.510$, which compares well with the theoretical value, $D/2 = 0.5$, given by equation (7).
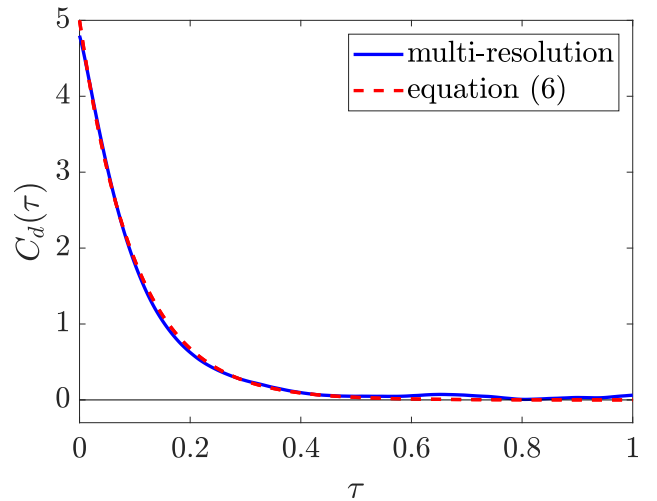

FIG. 6. *The velocity autocorrelation function for the multi-resolution model (blue solid line) for long-range interactions, estimated from long time simulation over dimensionless time of 10^3 dimensionless time units. It is compared with the result for the Langevin description of the whole dimer, given by equation (6) (red dashed line). The parameters are the same as in Figure 4, namely $\gamma = 10$, $D = 1$, $k = 10^6$, $M = 1$, $m = 10^{-3}$, $\ell_0 = 0.32$, together with $\bar{\omega} = 100$ and $N_1 = 10^5$.*

V. DISCUSSION AND CONCLUSIONS

In this paper, we have used two theoretical heat baths. Although these heat baths are based on qualitatively different descriptions of solvent-dimer interactions, they both lead to the Langevin description, given in equations (1)–(4), in a certain limit. In particular, we can use this limiting process to coarse-grain a part of the simulated dimer molecule, while use a detailed MD model to describe the rest of the molecule. Such a multi-resolution approach has potential to significantly speed up computer simulations of dynamics of macromolecules[38–42], provided that it is combined with additional multiscale and multi-resolution methodologies, discussed below.

Our long-range interaction model leads to the system of generalized Langevin equations, given by equations (39)–(40). Although we have worked in the parameter regime where the generalized Langevin equations can be well approximated by the system of Langevin equations given by (1)–(4), this will not be the case in other parameter regimes and for more realistic solvent descriptions, especially when the memory kernel is estimated from MD simulations[58,59]. One possible strategy in this case is to couple a detailed MD model with a stochastic coarse-grained model which is written with the help of additional variables[50–52]. To improve the efficiency of simulations further, one can then coarse-grain such a generalized Langevin description using a Brownian dynamics approach[14,60]. Brownian dynamics modelling can be further coupled with stochastic reaction-diffusion mod-

elling based on lattice-based (compartment-based) methods[22]. Lattice-based models are very attractive for simulations of intracellular processes, because they enable modelling of spatio-temporal processes in the whole cell or its significant part[61]. Coupling Brownian dynamics with compartment-based approaches has been used in a number of applications, including multi-resolution modelling of actin dynamics in filopodia[62,63] or for modelling intracellular calcium dynamics[64].

In this paper, we have investigated multi-resolution approaches, schematically described in Figure 1(a) and 1(b). Another class of multi-resolution approaches in the literature considers a fixed subdomain of the computational domain where a detailed modelling approach is used, which is coupled with a coarser model in the rest of the simulation domain[21,22]. Such an approach is useful, for example, when modelling intracellular ion dynamics. Ions pass through an ion channel in single file and an MD model has to be used to accurately compute the discrete, stochastic, current in the channel[65,66], while the details of the behaviour of individual ions are less important away from the channel where copy numbers may be very large. Thus, we can improve efficiency of our simulations if we allow ions to pass between regions with an explicitly modelled heat bath and a region where their trajectories are described by coarser stochastic models[51].

A similar multi-resolution approach can also be designed for our illustrative dimer model. It is schematically shown in Figure 1(c), where we identify the region with explicitly simulated heat bath as $\{x_1 > b\} = (b, \infty) \times \mathbb{R}^2$, where b is the fixed position of the boundary. We are again interested in the behaviour of the dimer in the MD model which would be considered in the full space, $\mathbb{R}^3$. However, we now want to replace solvent particles which are in $\{x_1 < b\} = (-\infty, b) \times \mathbb{R}^2$ by a coarser Langevin description (1)–(4). To do that, we have to carefully consider how we handle the transfer of monomers between $\{x_1 > b\}$ and $\{x_1 < b\}$. In Figure 1(c), we present a two-dimensional illustration of a monomer when it intersects the interface, $\{x_1 = b\}$. Such a monomer is subject to the collisions with heat bath particles on the part of its surface which lies in $\{x_1 > b\}$. This has to be compensated by using a suitable random force from $\{x_1 < b\}$, so that the overall model is equivalent to (1)–(4) in the Langevin limit. Such correction terms can be derived analytically for the case of a spherical monomer in our short-range interaction heat bath and are presented in References[14,54]. They can be used to couple the MD model with its corresponding Langevin description, which can be further coupled with Brownian dynamics, simulated using a much larger time step[14].

Mathematical analysis of multi-resolution methodologies can make use of the analysis of the model behaviour close to the boundaries of the computational domain. For example, derivations of reactive (Robin) boundary conditions of macroscopic models from their corresponding microscopic descriptions[67–69] can be generalized to the analysis of behaviour of molecules close to hybrid interfaces in multi-resolution schemes[21,30,31]. Analysis of open boundaries of MD schemes (i.e. boundaries which can transfer mass, momentum and energy) can lead to further understanding of multi-resolution schemes such as AdResS and hybrid continuum-particle dynamics[70], which enable efficient simulation of biomolecules at realistic physiological conditions[71].

Equations for coupled detailed/coarse-grained models can be systematically derived using Zwanzig's projection method, which has been used to address co-existence of atoms and beads (larger coarse-grained units) in the same dynamic simulations[72,73]. The equations of motion take the form of dissipative particle dynamics, which have been coupled with atomistic water simulations to design multi-resolution schemes in the literature[74]. Other multi-resolution methods couple atomistic water with specially designed coarse-grained water models[75] or with a continuum approach[35]. Coupling discrete and continuum approaches can also be done for different molecular species present in the system and our choice of a modelling approach for each species can be based on its relative abundance[76–78].

One of several important points which have been left out from our discussion is the discretization of time. Although our illustrative simulations use the same time step for both the MD model and the Langevin description, this is not the most efficient or desirable strategy, because the MD model requires much smaller time step than the corresponding Langevin equation. There is potential to design more efficient schemes by updating the coarser description only at certain multiples of the time step which is used in the most detailed model[39]. This is also the case when a modeller further coarse-grains the Langevin description into a Brownian dynamics model which uses even large timesteps[14].

Authors' contributions

Ravinda Gunaratne (RG) and Daniel Wilson (DW) wrote computer codes and peformed computer simulations to obtain illustrative results presented in Figure 3 (DW, RG), Figure 4 (DW, RG), Figure 5 (RG) and Figure 6 (RG). All authors worked on the analysis of both (short-range and long-range interaction) MD models, wrote the paper and gave final approval for publication.

Data accessibility

The computer codes used to compute illustrative results in Figures 3, 4, 5 and 6 have been uploaded as part of the supplementary material. Figures 1 and 2 contain no data.

REFERENCES

[1] D. Rapaport, *The Art of Molecular Dynamics Simulation* (Cambridge University Press, 2004).

[2] B. Leimkuhler and C. Matthews, *Molecular Dynamics*, Interdisciplinary Applied Mathematics, Vol. 39 (Springer, 2015).

[3] S. Marrink, H. Risselada, S. Yefimov, D. Tieleman, and A. de Vries, Journal of Physical Chemistry B **111**, 7812 (2007).

[4] S. Yesylevskyy, L. Schäfer, D. Sengupta, and S. Marrink, PLoS Computational Biology **6**, e1000810 (2010).

[5] S. Riniker and W. van Gunsteren, Journal of Chemical Physics **134**, 084110 (2011).

[6] L. Darré, M. Machado, P. Dans, F. Herrera, and S. Pantano, Journal of Chemical Theory and Computation **6**, 3793 (2010).

[7] J. Israelachvili, *Intermolecular and Surface Forces*, 3rd ed. (Academic Press, Elsevier, 2011).

[8] J. Rowlinson, *Cohesion: A Scientific History of Intermolecular Forces* (Cambridge University Press, 2002).

[9] J. Lennard-Jones, Proceedings of the Royal Society of London Series A **106**, 463 (1924).

[10] J. Lennard-Jones, Proceedings of the Physical Society **43**, 461 (1931).

[11] R. Holley, Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete **17**, 181 (1971).

[12] D. Dürr, S. Goldstein, and J. Lebowitz, Communications in Mathematical Physics **78**, 507 (1981).

[13] J. Dunkel and P. Hänggi, Physical Review E **74**, 051106 (2006).

[14] R. Erban, Proceedings of the Royal Society A **470**, 20140036 (2014).

[15] G. Ford, M. Kac, and P. Mazur, Journal of Mathematical Physics **6**, 504 (1965).

[16] R. Zwanzig, Journal of Statistical Physics **9**, 215 (1973).

[17] G. Bussi and M. Parrinello, Physical Review E **75**, 056707 (2007).

[18] D. Frenkel and B. Smit, *Understanding Molecular Simulation, From Algorithms to Applications*, 2nd ed. (Academic Press, Elsevier, 2002).

[19] D. Huggins, Journal of Chemical Physics **136**, 064518 (2012).

[20] P. Mark and L. Nilsson, Journal of Physical Chemistry A **105**, 9954 (2001).

[21] M. Flegg, J. Chapman, and R. Erban, Journal of the Royal Society Interface **9**, 859 (2012).

[22] M. Robinson, S. Andrews, and R. Erban, Bioinformatics **31**, 2406 (2015).

[23] M. Praprotnik, S. Matysiak, L. Delle Site, K. Kremer, and C. Clementi, Journal of Physics: Condensed Matter **19**, 292201 (2007).

[24] B. Ensing, S. Nielsen, P. Moore, M. Klein, and M. Parrinello, Journal of Chemical Theory and Computation **3**, 1100 (2007).

[25] M. Praprotnik, L. Delle Site, and K. Kremer, Journal of Chemical Physics **123**, 224106 (2005).

[26] R. Potestio, C. Peter, and K. Kremer, Entropy **16**, 4199 (2014).

[27] J. Zavadlav, M. Melo, S. Marrink, and M. Praprotnik, Journal of Chemical Physics **140**, 054114 (2014).

[28] J. Zavadlav, R. Podgornik, and M. Praprotnik, Journal of Chemical Theory and Computation **11**, 5035 (2015).

[29] J. Zavadlav, S. Bevc, and M. Praprotnik, European Biophysics Journal **46**, 821 (2017).

[30] M. Flegg, J. Chapman, L. Zheng, and R. Erban, SIAM Journal on Scientific Computing **36**, B561 (2014).

[31] M. Flegg, S. Hellander, and R. Erban, Journal of Computational Physics **289**, 1 (2015).

[32] M. Robinson, M. Flegg, and R. Erban, Journal of chemical physics **140**, 124109 (2014).

[33] C. Smith and C. Yates, Journal of The Royal Society Interface **15**, 20170931 (2018).

[34] B. Franz, M. Flegg, J. Chapman, and R. Erban, SIAM Journal on Applied Mathematics **73**, 1224 (2013).

[35] R. Delgado-Buscalioni, K. Kremer, and M. Praprotnik, Journal of Chemical Physics **131**, 244107 (2009).

[36] M. Machado, H. Gonzáles, and S. Pantano, Journal of Chemical Theory and Computation **13**, 51065116 (2017).

[37] E. Tarasova, V. Farafonov, R. Khayat, N. Okimoto, T. Komatsu, M. Taiji, and D. Nerukh, Journal of Physical Chemistry Letters **8**, 779 (2017).

[38] A. Fogarty, R. Potestio, and K. Kremer, Proteins **84**, 1902 (2016).

[39] E. Rolls, Y. Togashi, and R. Erban, Multiscale Modeling and Simulation **15**, 1672 (2017).

[40] E. Rolls and R. Erban, Journal of Chemical Physics **148**, 194111 (2018).

[41] N. Di Pasquale, R. Gowers, and P. Carbone, Journal of Computational Chemistry **35**, 1199 (2014).

[42] N. Di Pasquale and P. Carbone, Journal of Chemical Physics **146**, 084905 (2017).

[43] J. Dama, A. Sinitskiy, M. McCullagh, J. Weare, B. Roux, A. Dinner, and G. Voth, Journal of Chemical Theory and Computation **9**, 2466 (2013).

[44] C. Robert, Statistics and Computing **5**, 121 (1995).

[45] W. van Gunsteren and H. Berendsen, Molecular Physics **45**, 637 (1982).

[46] W. Wang and R. Skeel, Molecular Physics **101**, 2149 (2003).

[47] A. Brunger, C. Brooks, and M. Karplus, Chemical Physics Letters **105**, 495 (1984).

[48] N. Gronbech-Jensen and O. Farago, Molecular Physics **111**, 983 (2013).

[49] K. Burrage, I. Lenane, and G. Lythe, SIAM Journal on Scientific Computing **29**, 245 (2007).

[50] A. Davtyan, J. Dama, G. Voth, and H. Andersen, Journal of Chemical Physics **142**, 154104 (2015).

[51] R. Erban, Proceedings of the Royal Society A **472**, 20150556 (2016).

[52] A. Davtyan, G. Voth, and H. Andersen, Journal of Chemical Physics **145**, 224107 (2016).

[53] B. Leimkuhler, Noorizadeh, and Theil, Journal of Statistical Physics **135**, 261 (2009).

[54] R. Erban and S. J. Chapman, *Stochastic modelling of reaction-diffusion processes* (Cambridge University Press, ISBN 9781108498128, 2019).

[55] R. Miller and E. Tadmor, Modelling and Simulation in Materials Science and Engineering **17**, 053001 (2009).

[56] I. Kevrekidis, C. Gear, J. Hyman, P. Kevrekidis, O. Runborg, and K. Theodoropoulos, Communications in Mathematical Sciences **1**, 715 (2003).

[57] E. Biyikli, Q. Yang, and A. To, Computer Methods in Applied Mechanics and Engineering **274**, 42 (2014).

[58] G. Jung, M. Hanke, and F. Schmid, Journal of Chemical Theory and Computation **13**, 2481 (2017).

[59] H. Shin, C. Kim, P. Talkner, and E. Lee, Chemical Physics **375**, 316 (2010).

[60] R. Erban and S. J. Chapman, Physical Biology **6**, 046001 (2009).

[61] E. Roberts, J. Stone, and Z. Luthey-Schulten, Journal of Computational Chemistry **34**, 245 (2013).

[62] R. Erban, M. Flegg, and G. Papoian, Bulletin of Mathematical Biology **76**, 799 (2014).

[63] U. Dobramysl, G. Papoian, and E. Erban, Biophysical Journal **110**, 2066 (2016).

[64] U. Dobramysl, S. Rüdiger, and R. Erban, Multiscale Modelling and Simulation **14**, 997 (2016).

[65] T. Allen, S. Kuyucak, and S. Chung, Biophysical Chemistry **86**, 1 (2000).

[66] M. Jensen, D. Borhani, K. Lindorff-Larsen, P. Maragakis, V. Jogini, M. Eastwood, R. Dror, and D. Shaw, Proceedings of the National Academy of Sciences USA **107**, 5833 (2010).

[67] R. Erban and S. J. Chapman, Physical Biology **4**, 16 (2007).

[68] R. Erban and S. J. Chapman, Physical Review E **75**, 041116 (2007).

[69] J. Chapman, R. Erban, and S. Isaacson, SIAM Journal on Applied Mathematics **76**, 368 (2016).

[70] J. Delgado-Buscalioni, R.and Sablić and M. Praprotnik, European Physical Journal Special Topics **224**, 2331 (2015).

[71] J. Zavadlav, J. Sablic, R. Podgornik, and M. Praprotnik, Bio-

physical Journal **114**, 2352 (2018).

[72] P. Español, Europhysics Letters **88**, 40008 (2009).

[73] N. Di Pasquale, T. Hudson, and M. Icardi, "Systematic derivation of hybrid coarse-grained models," (2018), available as https://arxiv.org/abs/1804.08157.

[74] J. Zavadlav and M. Praprotnik, Journal of Chemical Physics **147**, 114110 (2017).

[75] H. Gonzalez, L. Darré, and S. Pantano, Journal of Physical Chemistry B **117**, 14438 (2013).

[76] Z. Liu, Y. Pu, F. Li, C. Shaffer, S. Hoops, J. Tyson, and Y. Cao, Journal of Chemical Physics **136**, 034105 (2012).

[77] A. Duncan, R. Erban, and K. Zygalakis, Journal of Computational Physics **326**, 398 (2016).

[78] B. Franz and R. Erban, in *Dispersal, individual movement and spatial ecology: A mathematical perspective*, edited by M. Lewis, P. Maini, and S. Petrovskii (Springer, 2012).