

Real-time temperature measurement in stochastic rotation dynamics

Fan, Rong; Zachariah, Githin T.; Padding, Johan T.; Hartkamp, Remco

DOI

[10.1103/PhysRevE.104.034124](https://doi.org/10.1103/PhysRevE.104.034124)

Publication date

2021

Document Version

Final published version

Published in

Physical Review E

Citation (APA)

Fan, R., Zachariah, G. T., Padding, J. T., & Hartkamp, R. (2021). Real-time temperature measurement in stochastic rotation dynamics. *Physical Review E*, 104(3), Article 034124. <https://doi.org/10.1103/PhysRevE.104.034124>

Important note

To cite this publication, please use the final published version (if applicable). Please check the document version above.

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights. We will remove access to the work immediately and investigate your claim.

Real-time temperature measurement in stochastic rotation dynamicsRong Fan,^{*} Githin T. Zachariah^{⊕,†}, Johan T. Padding^{⊕,‡}, and Remco Hartkamp[§]*Complex Fluid Processing, Process and Energy Department, Delft University of Technology, 2628 CB Delft, The Netherlands*

(Received 9 July 2021; accepted 3 September 2021; published 20 September 2021)

Many physical and chemical processes involve energy change with rates that depend sensitively on local temperature. Important examples include heterogeneously catalyzed reactions and activated desorption. Because of the multiscale nature of such systems, it is desirable to connect the macroscopic world of continuous hydrodynamic and temperature fields to mesoscopic particle-based simulations with discrete particle events. In this work we show how to achieve real-time measurement of the local temperature in stochastic rotation dynamics (SRD), a mesoscale method particularly well suited for problems involving hydrodynamic flows with thermal fluctuations. We employ ensemble averaging to achieve local temperature measurement in dynamically changing environments. After validation by heat diffusion between two isothermal plates, heating of walls by a hot strip, and by temperature programmed desorption, we apply the method to a case of a model flow reactor with temperature-sensitive heterogeneously catalyzed reactions on solid spherical catalysts. In this model, adsorption, chemical reactions, and desorption are explicitly tracked on the catalyst surface. This work opens the door for future projects where SRD is used to couple hydrodynamic flows and thermal fluctuations to solids with complex temperature-dependent surface mechanisms.

DOI: [10.1103/PhysRevE.104.034124](https://doi.org/10.1103/PhysRevE.104.034124)**I. INTRODUCTION**

Many physical and chemical processes involve an exchange of energy with the surroundings or the conversion of one form of energy to another. One of these forms is thermal energy. Temperature is commonly perceived as the “hotness” or “coldness” of matter [1]. More precisely, from statistical mechanics we know that temperature can be expressed in terms of the average kinetic energy of the constituent atoms in the absence of macroscopic flow [2].

Temperature is important for various physical and chemical processes, including adsorption, desorption, and chemical reactions, which often take place at interfaces, such as the interface between a solid catalyst and a fluid containing the reactants and products. The rates with which these processes take place are not only influenced by microscale kinetic factors but also by local macroscale properties such as pressure, hydrodynamic velocity fields, and, importantly, temperature. Applications of these processes often involve fluids in complex geometries, for example adsorption in porous media or heterogeneously catalyzed reactions taking place in packed bed reactors. When modeling such systems, one usually encounters the problem of having to tackle phenomena at different length scales ranging from the microscale to the macroscale.

Macroscale computational techniques are based on the continuum assumption. These techniques generally require

solving conservation equations in the form of partial differential equations. As a result, the continuum methods model macroscopic effects of the embedded microscopic behavior directly [3]. The continuum assumption breaks down for systems where molecular effects such as Knudsen diffusion, and other effects that involve finite numbers of particles, are prominent. In such a case, microscale techniques, such as molecular dynamics (MD) simulations are preferred. However, microscale techniques are usually too computationally expensive to simulate larger volumes of continuum fluid phases.

In such cases, it is advantageous to use mesoscale simulation methods, such as lattice Boltzmann, direct simulation Monte Carlo, dissipative particle dynamics (DPD), or stochastic rotation dynamics (SRD) [4]. The advantage of SRD in particular is that it automatically accounts for thermal fluctuations and hydrodynamics, and can easily be extended to include reactions on surfaces. This makes it a promising tool to investigate the coupling of convection-diffusion mechanisms to microkinetic (adsorption-reaction-desorption) phenomena taking place on reactive surfaces and in porous media.

In SRD, a system consisting of point particles is evolved with a discrete streaming step and a collision step in which all particles in a collision cell simultaneously exchange momentum. This general approach is often referred to as multiparticle collision dynamics (MPCD or MPC) [4] and shares some features with the direct simulation Monte Carlo approach. SRD refers to a particular implementation in which the collisions are executed through a stochastic rotation of the relative velocities of the particles in the collision cell [5]. Ihle and Kroll [6] pointed out that the introduction of a grid-shift procedure is necessary to sustain Galilean invariance. The fact

^{*}r.fan@tudelft.nl[†]githin.zachariah@ul.ie[‡]j.t.padding@tudelft.nl[§]r.m.hartkamp@tudelft.nl

that SRD is particle based and generates Navier-Stokes hydrodynamics in an efficient way makes this method very suitable for simulations of complex systems such as equilibrium and nonequilibrium colloid and polymer solutions [7–18], microswimmers [19–25], and viscoelastic fluids [26–28]. Applications of SRD to biological systems have sprung up, such as for biological functional molecules, bacteria, and cell suspensions and their dynamic behaviors [29–34].

Most of the implementation cases of SRD are assumed to be isothermal [35,36]. However, recently researchers start to pay attention on the nonequilibrium simulations with SRD. Lepri [37] studied nonequilibrium steady states of a one-dimensional fluid in a finite domain with thermal walls. Investigating temperature effects has not been the first concern in this method and is complicated by the feature of inherent thermal fluctuations. More precisely, in MPC or SRD, the temperature is not set as a direct parameter but is inherent in the particles' velocity fluctuations. This feature brings difficulties for measuring and simulating real-time local temperature change in the system, since the temperature is not an explicitly accessible variable of the simulation. Usually when there is viscous heating present in a nonequilibrium simulation, a thermostat is used to regulate temperature and establish a steady state. A thermostat rescales the relative velocities of particles in a collision cell, without compromising the cell-averaged velocity, nor the Maxwell-Boltzmann distribution in the collision step [5]. Thermostats are widely used in MPC, for example, Anderson thermostat [35]. Dynamic parameter values remain constant in thermostatted systems, while the kinetic temperature relaxes to a fixed temperature.

If certain system properties are highly temperature dependent, then local temperature variations can lead to strong feedback effects. An important example of this is heterogeneous catalysis, where the rates of catalyzed chemical reactions can change orders of magnitude due to a relatively small change of temperature. Because of the high sensitivity of reaction rates to local temperature, the conversion and selectivity in structured or packed bed reactors is intricately linked to variations in temperature that can change over relatively small time and length scales. Modelling such reactors with proper temperature-dependent local reaction rates thus requires measurement of the local temperature in real time.

In this work, we investigate a novel way to measure and simulate thermal dynamics in SRD, by which we extend the field of possible application of the SRD method. This paper is arranged as follows. In Sec. I, we provide a theoretical background of SRD for modeling hydrodynamics and a newly developed technique for real-time local temperature measurement. In Sec. III, we validate the method for different cases of heat conduction through a fluid phase and between a fluid and a solid phase and investigate heat effects in systems with surface reactions. Finally, in Sec. IV we give our conclusions and outlook.

II. METHODOLOGY

A. Fluid model

Stochastic rotation dynamics is a mesoscale simulation method in which the fluid (either a gas or a liquid) is

coarse-grained to a set of point particles that interact according to highly efficient multiparticle collision rules [5,15]. The resulting dynamics obeys the Navier-Stokes equations at larger scales, while also offering the advantage of automatically accounting for thermal fluctuations and (coarse-grained) molecular diffusion. N particles are placed in the system to represent the fluid, with a total mass of $\sum_{j=1}^k X_j m_j N$. Here X_j , m_j , and k are the mole fraction, mass of component j , and the number of components in the system, respectively. In between two subsequent collisions, in the so-called streaming step, the position of each particle i is updated by a first-order Euler scheme. If \mathbf{r}_i^t and \mathbf{v}_i^t are the position and velocity vectors of particle i at time t , and Δt_s is the timestep used during streaming, then the new position vectors are given by:

$$\mathbf{r}_i^{t+1} = \mathbf{r}_i^t + \mathbf{v}_i^t \Delta t_s. \quad (1)$$

If there is a body force applied to the fluid, e.g., to drive a flow, then the velocity of particle i is updated in the streaming step according to

$$\mathbf{v}_i^{t+1} = \mathbf{v}_i^t + \mathbf{g} \Delta t_s. \quad (2)$$

In this expression, \mathbf{g} is the acceleration associated with the body force.

Particles only interact with each other during the collision step. To achieve this in a computationally efficient way, space is coarse-grained every collision time interval Δt_c (chosen as an integer multiple of Δt_s) into a grid of cubic lattice cells, with a size of a_0 in each direction. The particle interactions are modelled by instantaneous exchange of their momenta in each lattice cell. In SRD, the velocity vector of each particle is updated according to:

$$\mathbf{v}_i' = \bar{\mathbf{v}} + \mathbf{\Omega}(\mathbf{v}_i - \bar{\mathbf{v}}), \quad (3)$$

where \mathbf{v}_i indicates the postcollision velocity. In the above expression, $\bar{\mathbf{v}} = \sum_j m_j \mathbf{v}_j / (\sum_j m_j)$ is the center-of-mass velocity of all particles j in the same cell as particle i and $\mathbf{\Omega}$ is a matrix that rotates the velocities by a fixed angle α around an axis (where for each cell a different randomly oriented axis is chosen, leading to the stochasticity of the method). This rotation step leads to mixing of the particle velocities, i.e., viscous dissipation. The local conservation of mass and momentum on the scale of a collision cell leads to hydrodynamic behavior on larger scales, in agreement with the Navier-Stokes equations.

We note that a random grid shift procedure is needed before each collision step to ensure Galilean invariance [6], wherein the SRD grid is shifted randomly in all three Cartesian directions at every collision step, with the same shift applied to every SRD cell. While fixing the Galilean invariance of the fluid, the shifted grids can coincide with the domain walls, resulting in partially occupied cells at the walls, leading to a nonzero slip velocity [5,38,39]. A correction is made by adding ‘‘ghost’’ particles at the wall [40,41], with velocities satisfying the Maxwell-Boltzmann distribution with zero mean velocity and a variance corresponding to the fluid temperature [5].

In our simulations, the units of length, mass, and energy are chosen equal to a_0 (the length of a collision cell), m_0 (the mass of the majority species SRD particle), and $k_B T_0$ (thermal energy at the reference temperature T_0), where k_B is Boltzmann's constant, leaving the average number of particles per collision

TABLE I. Simulation parameters of SRD and derived units. In our simulations the collision cell size, majority species mass, and thermal energy at the reference temperature are the units of length, mass, and energy, respectively.

SRD fluid simulation parameters	
$a_0 \equiv 1$	Collision cell size
$m_0 \equiv 1$	Solvent (majority species) mass
$k_B T_0 \equiv 1$	Thermal energy (at reference temperature)
γ	Average number of particles per collision cell
Δt_s	Streaming integration timestep
Δt_c	Collision time interval
α	SRD rotation angle
Derived units	
Density	m_0/a_0^3
Time	$t_0 = a_0 \sqrt{\frac{m_0}{k_B T_0}}$
Diffusion coefficient	$D_0 = \frac{a_0^2}{t_0} = a_0 \sqrt{\frac{k_B T_0}{m_0}}$

cell γ , the time interval between successive collisions Δt_c , and the rotation angle α as independent simulation parameters. The units of other variables, such as density, time, and various diffusion coefficients, can be derived from the three base units, as given in Table I.

An important advantage of SRD is that the transport properties are well defined for arbitrary values of the simulation parameters. The simplified dynamics of SRD allows derivation of analytical expressions for most transport properties [42–45], which is not possible for other particle-based methods such as MD and DPD. The streaming step and collision step lead to separate (and additive) contributions to the transport coefficients, usually referred to as kinetic and collisional contributions (see Table II). Analytical expressions for the (kinematic) viscosity ν and thermal diffusivity D_T have been derived and validated by Ihle and Tüzel [43,46,47]. The self-diffusion coefficient of SRD particles has been calculated and investigated by Kapral [4,48]. In contrast to the shear viscosity and thermal diffusivity, there is no analytical expression for the collisional contribution to the self-diffusion coefficient be-

TABLE II. Kinetic (streaming) and collisional contributions to SRD transport coefficients for a three-dimensional single-component SRD fluid, valid up to order $1/\gamma^2$. Note that the shear viscosity is expressed as a kinematic viscosity, i.e., with the same units as thermal diffusivity and self-diffusivity.

	Kinetic contribution $\times k_B T \Delta t_c / (2m)$
Shear viscosity	$\nu = \frac{5\gamma}{(\gamma-1+e^{-\gamma})[2-\cos\alpha-\cos(2\alpha)]} - 1$
Thermal diffusivity	$D_T = \frac{3}{1-\cos\alpha} \left(1 - \frac{1}{\gamma}\right) + \frac{24}{5\gamma} - 1$
Self-diffusivity	$D_s = \frac{3\gamma}{(\gamma-1+e^{-\gamma})(1-\cos\alpha)} - 1$
	Collisional contribution $\times a_0^2 / \Delta t_c$
Shear viscosity	$\nu = \frac{1}{18\gamma} (\gamma - 1 + e^{-\gamma}) [1 - \cos\alpha]$
Thermal diffusivity	$D_T = \frac{1}{15\gamma^2} (\gamma - 1) [1 - \cos\alpha]$

cause it has proven difficult to theoretically include effects of hydrodynamic correlations on the self-diffusivity. The analytical expression for self-diffusivity is, therefore, most accurate for $\lambda > 0.6$, where $\lambda = \Delta t_c / t_0$ is the dimensionless mean free path. For such a large mean free path, the random collision approximation can be expected to be valid.

B. Temperature change and measurement

1. Measuring temperature in the fluid phase

In SRD, similarly to molecular dynamics simulations, temperature can be calculated from the average fluctuation velocities, i.e., particle velocities minus any contribution due to convective motion, of a sample of particles [49]. According to statistical mechanics, the instantaneous temperature of a collection of n particles may be determined from:

$$k_B T = \frac{1}{3(n-1)} \sum_{i=1}^n m_i (\mathbf{v}_i - \bar{\mathbf{v}})^2, \quad (4)$$

where the number of degrees of freedom (per Cartesian direction) n is reduced by one to account for the constraint that the average of the velocities \mathbf{v}_i must equal the center-of-mass velocity $\bar{\mathbf{v}}$ of all n particles. Based on Eq. (4), a local and instantaneous temperature $T(\mathbf{x}_c, t)$ can be determined for each cell located at \mathbf{x}_c . However, the number of particles per cell (typically in the range of 5 to 20) is usually too low to perform a statistically meaningful local and instantaneous measurement. For a system in steady state, a proper solution to find the—possibly spatially dependent—temperature is to locally average over many subsequent time instances, in other words to perform a long time average. Conversely, for systems with a certain symmetry in the geometry, this symmetry may be exploited to average over multiple equivalent locations and find the—possibly time dependent—average temperature. However, an unsteady-state system with a complex geometry, such as the start-up of reactive flow in structured or packed bed reactors, calls for an innovative way of averaging that preserves the local and instantaneous nature of the temperature field.

In this work we propose to determine the local and instantaneous temperature by averaging over sufficiently many independent realisations of the same system, i.e., replica averaging.

There are two ways in which replica averaging can be implemented: serial and parallel. In the serial implementation, each simulation is run to completion and saved and then reinitialized and repeated. The accumulated solution is then averaged at the end of n_s iterations. In a parallel implementation, n_s versions (replicas) of the simulation are performed simultaneously, while exchanging data and averaging between the replicas at each time step.

Both implementations have their advantages and disadvantages. Serial implementation is simpler to realize. It is well suited for problems in which the dynamics are independent of on-the-fly information about the local temperature. For example, a study of heat transfer through a fluid between boundaries at different temperature can be easily realized with this technique since the influence of local temperature variations on the heat transfer is emergent. Parallel implementation

is more involved, but it allows us to solve problems where on-the-fly information about the local temperature influences the dynamics of the system. For example, determining the reaction rate of a heterogeneously catalyzed reaction requires access to a statistically meaningful measurement of the local instantaneous temperature. By using the power of parallel computing we can implement an ensemble of n_s equivalent versions of the system, while exchanging data and averaging each time step.

SRD or MPCD is naturally suitable for parallel computing, owing to the fact that the system is discretized in independent particles, and no differential equations need to be solved to simulate the fluid. Howard *et al.* exploited the parallel computational capabilities of GPUs in the implementation of MPCD algorithm [50]. One way to implement parallel SRD is by using Compute Unified Device Architecture, an extension to the C language developed by Nvidia. The advantage of this technique is that it can accelerate the simulation even if there is only one instance running. The implementation is suitable to all potential applications of SRD. However, such GPU-based implementations are often bottlenecked by hardware limitations. GPUs often possess smaller RAM capacities than CPUs, which could be limiting for a memory-intensive method such as the one studied in this work.

In this work, a new variation of SRD is developed that employs Open MPI. Parallel replicas of the simulation are executed simultaneously. Communication between the parallel simulations is only needed when averaging macroscopic properties such as local temperature. This can be handled efficiently in Open MPI. The parallel algorithm used in this study is shown in Fig. 1. The simulation starts by initializing multiple instances (four instances in Fig. 1) of the simulation simultaneously. For each time step, every instance undergoes, independently from other instances, the streaming, wall bounce-back, and collision step, and any chemical or physical process implemented in the simulation that possibly partially depends on the local macroscopic variables. Local macroscopic variables are evaluated for each instance and then averaged between the instances and delivered back to each node. The overall mean values of macroscopic variables can then be used in the next time step.

2. Heat exchange between bulk fluid and bulk solid phase

The temperature of the bulk of a solid phase is often neglected in SRD. Rather, the wall temperature is usually accounted for via boundary conditions (BCs). In SRD, the most commonly used BC at a wall is the bounce back BC, leading to an effective no-slip boundary condition. In bounce back, all components of a particle's velocity are inverted when it collides with a solid surface or a hard wall of the domain, as in Eq. (5),

$$\mathbf{v}_i \rightarrow -\mathbf{v}_i. \quad (5)$$

However, the bounce back BC does not allow for energy exchange between the fluid and the solid since the distribution of the postcollisional velocity exactly mirrors the distribution of the precollisional velocity. In other words, from a thermal perspective bounce back BCs are equivalent to adiabatic boundaries.

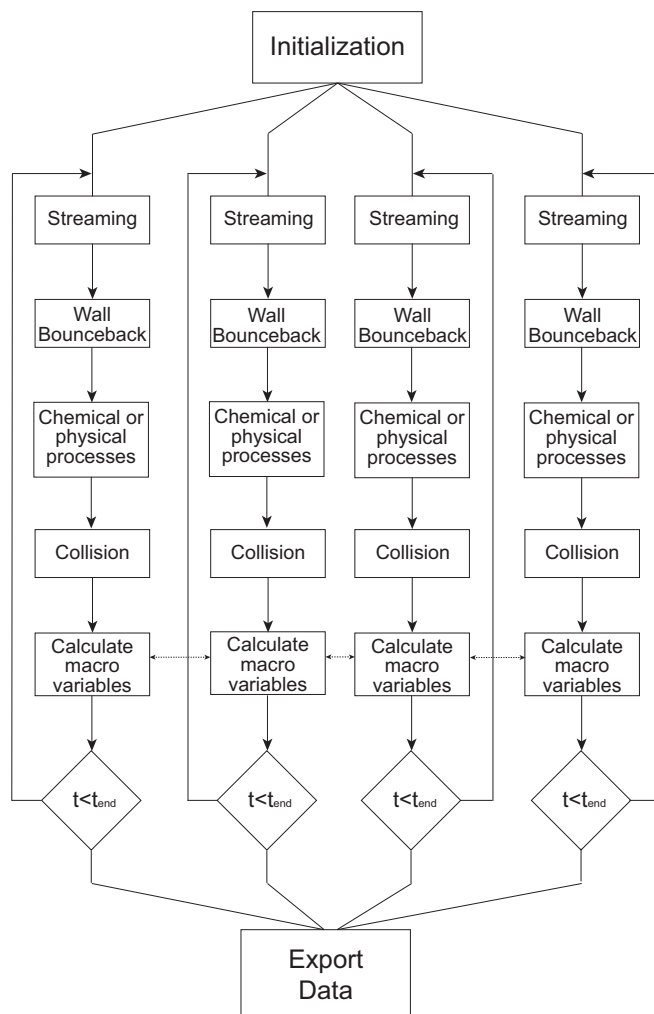


FIG. 1. Flowchart of the OpenMPI algorithm for parallel averaging of local macroscopic variables in SRD. In this example we show $n_s = 4$ instances (replicas).

Any system with viscous fluid flow exhibits viscous dissipation: the kinetic energy associated with fluid flow is converted into thermal energy, which is observed as an increase in velocity fluctuations. Previous SRD studies [38,51] removed this excess heat by enforcing a predetermined temperature. Such a thermostat is not utilizable here since the goal of this work is to mimic systems in which reaction heat or viscous dissipation can cause the temperature to evolve in time and space.

In many real systems, viscous heat generated in the fluid is removed via the confining walls. To mimic this, an option is to use the stochastic BC suggested by Padding and Louis [15]. In this BC, postcollision velocities at the surface are generated from the expected velocity distribution of SRD particles, at the desired wall temperature. This method results in a small slip velocity in systems with flow tangential to the wall, because of the nonzero average tangential velocity before the collision with the wall. A correction to this was proposed by Bolinteanu *et al.* [40]. Essentially, the Gaussian distribution in the tangential direction is biased with the local mean velocity to counteract the slip. This leads to the following probabilities

to generate new velocity components v_{ni} and v_{ti} normal and tangential to the wall, respectively, for a particle i that bounced into the wall:

$$P(v_{ni}) = v_{ni} \exp\left(-\frac{m_i v_{ni}^2}{k_B T_w}\right) \Theta(v_{ni}), \quad (6)$$

$$P(v_{ti}) = \sqrt{\frac{k_B T_w}{2\pi m_i}} \exp\left[-\frac{m_i (v_{ti} + \bar{v}_{ti})^2}{k_B T_w}\right], \quad (7)$$

where T_w is the wall temperature, $\Theta(v_{ni})$ is a Heaviside function ensuring that only positive normal velocities are generated, and \bar{v}_{ti} is the fluid flow mean particle velocity of the cell in which particle i was located in the previous time step. Note that the $+$ sign in front of \bar{v}_{ti} in Eq. (7) is not a typo, because the mean tangential postcollisional velocity should be equal to $-\bar{v}_{ti}$, which corrects the slip velocity by using a bounce-back scheme on the mean local tangential velocity. No additional noise is introduced, since the mean velocity profile is obtained by averaging. Even though the number of particles in a single SRD cell is limited, the correction of the boundary condition is not affected. This method will henceforth be called the biased stochastic boundary condition.

The biased stochastic bounce-back BC not only enables us to specify the temperature of the wall boundary but also to describe the heat flux through the boundary between the fluid and solid phase. To this end it is important to also consider thermal conduction through the solid structure.

Temperature conduction in a solid is described by the governing energy equation:

$$\rho_s C_s \frac{\partial T}{\partial t} = K_s \nabla^2 T, \quad (8)$$

where ρ_s is the density of the solid material, C_s is the heat capacity (per unit mass), and K_s is the thermal conductivity of the solid. Note that in practice we specify the thermal diffusivity $K_s/(\rho_s C_s)$ (in units of D_0) and the heat capacity per unit volume $\rho_s C_s$ (in units of $k_B T_0/a_0^3$) of the solid phase.

In this work the temperature field in a solid is evaluated on a Cartesian grid. Solids with curved surfaces are therefore approximated into cuboidal volumes. However, the fluid phase in coarse-grained particle form is not influenced by this approximation. The spatial derivatives on the right-hand side of Eq. (8) can be discretized using a central difference scheme.

For boundary cells, there are no more solid cells in one or more directions. Special care has to be taken in the directions where the solid is contacting the fluid. Taking a one-dimensional case, we can discretize the spatial derivative as:

$$\begin{aligned} K_s \frac{\partial^2 T}{\partial x^2} &= \frac{K_s \left(\frac{\partial T}{\partial x}\right)_{x+\Delta x/2} - K_s \left(\frac{\partial T}{\partial x}\right)_{x-\Delta x/2}}{\Delta x} \\ &= \frac{-F_{x+} + F_{x-}}{\Delta x}, \end{aligned} \quad (9)$$

where F_{x+} and F_{x-} denote the heat flux at the right and left interfaces of the cell. For interior cells, each of these terms can be further discretized with a general central difference scheme. For boundary cells, the flux term on the interface contacting the wall is replaced with the actual flux from the SRD fluid, which is determined as follows.

In the case of arbitrary geometries, a solid temperature mesh spans across the entire simulation domain, indicating cells that contain solid with a flag. The solid cells in contact with the fluid can be classified as a boundary cell, cells in the fluid as fluid cells, and solid cells fully surrounded by other solid cells as interior cells. Each solid temperature mesh cell contains the coefficients for itself and its six closest neighbors. In the preprocessing step, the interior cells are identified and given their coefficients. Then, the boundary cells are identified and given their coefficients according to Eq. (9), leaving out the external flux terms. This allows us to simply calculate the total energy transfer due to all particle-wall collisions in the cell without having to store in memory every location of surface collision.

When a particle collides with a wall, the exact location of the contact is calculated and the corresponding solid temperature mesh cell is identified. The flux due to the collision is then added to the flux variable of the mesh cell. All interior wall cells are given a flux value of 0 by default. The solid temperatures are then updated using the coefficient matrix to determine the discretization and the flux variable to add the necessary flux from the SRD fluid. This ensures that all wall cells can be updated with the same code, regardless of its contact with the fluid. A drawback of this method is the additional memory required to store the coefficient matrices. However, this memory requirement is small compared to that required to save the large number of simultaneous simulations.

The wall temperature T_w is determined from the solid cell temperature closest to the interface, and the postbounce-back velocity of the particle is calculated using Eqs. (6) and (7). This essentially changes the temperature of the outgoing particles to that of the local wall. The particle collisions with the interface of a specific solid cell during a time step result in energy change

$$\Delta E = \sum_i \frac{1}{2} m_i (\Delta v_{xi}^2 + \Delta v_{yi}^2 + \Delta v_{zi}^2), \quad (10)$$

that quantifies the heat flux into the solid cell:

$$F_t = \frac{\Delta E}{\Delta x^2 \Delta t}. \quad (11)$$

This heat flux is used in Eq. (9) to update the solid temperature [52]. This creates a two-way exchange of energy between the solid and the fluid.

3. Surface heat sources

In the previous part, we dealt with energy exchange between a fluid and a solid. We now consider the case where thermal energy is generated *on* the interface, for example as a result of adsorption or a (heterogeneous) surface reaction. Since the surface has no volume, and therefore no heat capacity, the question is how this heat should be initially distributed between the fluid phase and solid phase. For materials used in most common applications (solids in the form of metals, fluids in the form of water, air, or other gases), the thermal diffusivity in the solid is larger than that in the fluid. In that case, surface heat tends to first predominantly transfer to the solid phase and only then to the fluid phase. This is the approach taken in this work.

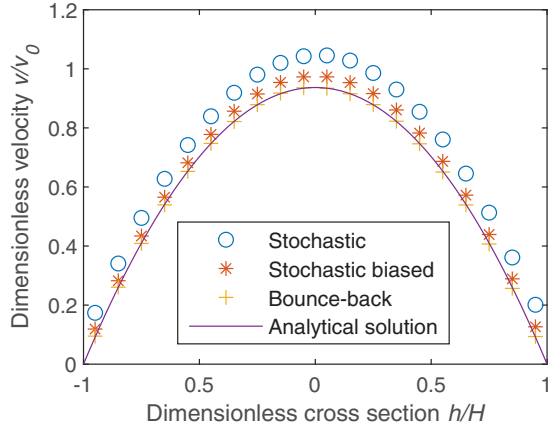


FIG. 2. Comparison of analytical and simulated velocity in Poiseuille flow. The walls are kept at a constant temperature.

When there is a surface heat source, the energy change in Eq. (10), which is used to determine the heat flux into the solid, must be replaced by:

$$\Delta E = \sum_i \frac{1}{2} m_i (\Delta v_{xi}^2 + \Delta v_{yi}^2 + \Delta v_{zi}^2) + \Delta E_{\text{surf}}, \quad (12)$$

where ΔE_{surf} is the amount of heat generated on the surface.

III. RESULTS

Before investigating a complex process involving heat and mass transport, we first validate the method by computing the velocity profile under the modified boundary condition, the heat diffusion between two isothermal plates, and between a hot strip and solid walls that can be heated. We then present the temperature measurement during programed desorption and a chemical reaction involving both the fluid phase and the solid phase.

A. Biased stochastic boundary condition

We validate the boundary conditions by comparing the simulated flow profile and the analytical solution for a planar Poiseuille flow. The SRD fluid parameter values are chosen as: $\gamma = 7$, $\Delta t_c = 0.5$, and $\alpha = 90^\circ$. The flow is induced by a body force $g = 0.001$ (in SRD units) acting on the SRD particles. This yields a Reynolds number of approximately 10.

The velocity profiles in Fig. 2 show that the bounce-back BC yields the closest agreement with the analytical solution. As expected, the stochastic BC shows some amount of slip, which is mostly corrected by the biased stochastic BC in Eq. (7). In this work, to demonstrate the significance of real-time temperature measurement and heat-transfer between solid and fluid, the biased stochastic BC is chosen in further simulations.

B. Heat diffusion between isothermal plates

Heat transfer is tested by simulating a fluid confined between two infinite plates (i.e., periodic boundary condition are applied in the x and z directions). The plates are located at

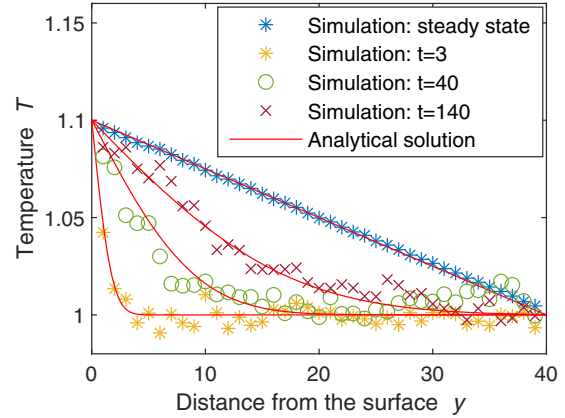


FIG. 3. Time evolution of the temperature distribution between two isothermal plates; the analytical solution is under the assumption of a constant thermal diffusivity evaluated at the temperature of the hot plate.

$y = 0$ and $y = L_y$, where L_y is the distance between the plates. In this case, the geometry of the simulation domain is set as $L_x = 15$, $L_y = 40$, and $L_z = 5$. The wall at $y = L_y$ is maintained at the initial temperature of 1, while the wall at $L = 0$ is instantaneously heated to a temperature of 1.1 at time $t = 0$, and maintained at this temperature throughout the simulation. The fluid number density was chosen to be $\gamma = 7$, the rotation angle $\alpha = 90^\circ$, and time step size $\Delta t_c = 0.5$. Following the equations in Table II, the thermal diffusivity varies between 0.5806 and 0.6371 depending on temperature, if the average fluid density of $\gamma = 7$ can be assumed fixed. 7500 simulations were conducted simultaneously for ensemble averaging. Analytically, one-dimensional heat transfer from a plate to an infinite domain is described by the following expression:

$$\frac{\partial T}{\partial t} = \frac{\partial}{\partial y} \left(D_T \frac{\partial T}{\partial y} \right), \quad (13a)$$

$$T(y, t = 0) = T_0 \quad T(y = 0, t) = T_w, \quad (13b)$$

where T_0 and T_w are the initial system temperature and the heated wall temperature, respectively. If the thermal diffusivity D_T is assumed to be constant, then the solution to the above differential equation can be readily found to be the complementary error function, with a width growing in time as $\sqrt{D_T t}$.

In the simulation, the existence of the cold wall at $y = L_y$ invalidates this solution for an infinite domain, except for the initial times of the simulation when the effect of the hot wall has not yet reached the opposite cold wall. Moreover, the thermal diffusivity is not constant in the simulation, because it depends on the local temperature and local number density. For relatively small temperature differences between the two walls we can approximate the thermal diffusivity to be constant at the value of the hot wall (which is most relevant for thermal diffusion of the hotter parts of the fluid). Following this approach, Fig. 3 shows that the temperature evolution follows the expected analytical solution.

The thermal diffusivity can also be measured from the SRD simulation in various ways [6,42,45,47,53,54]. For example, Ihle *et al.* [43] used a discrete-time projection operator tech-

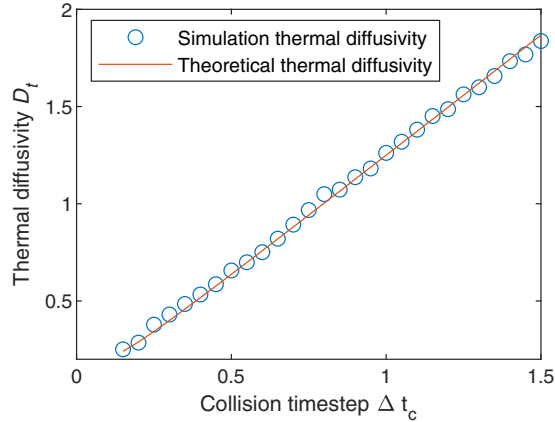


FIG. 4. Comparison between the theoretical thermal diffusivity (Table II) and the thermal diffusivity obtained from the measured heat flux [Eq. (15)].

nique to obtain the Green-Kubo relations for the transport coefficients. With the establishment of real-time temperature measurement in the present study, the thermal diffusivity can be measured more directly and computationally efficiently from the steady-state heat flux to the walls during conduction between flat plates, given by:

$$F_{t,x=0} = K \frac{dT}{dx}, \quad (14)$$

$$D_T = \frac{K}{\rho C_p}, \quad (15)$$

where $F_{t,x=0}$ denotes the thermal flux at $x = 0$ (or any other plane in steady state), K is the thermal conductivity of the fluid, ρ the mass density, and C_p the heat capacity per unit mass. We apply this method to a system with fluid particle density $\rho = m\gamma = 50$. The resulting thermal diffusivity is shown in Fig. 4 for various settings of the collision time step Δt_c and compared with the theoretical solution for thermal diffusivity derived in the work of Ihle *et al.* [43]. The simulated results closely match the analytical solution.

C. Fluid confined between planar walls

The exchange of heat between solid and fluid phases plays a key role in many processes. For example, the reaction kinetics and the catalyst efficiency in a heterogeneous catalytic reactor can be sensitive to temperature, and the formation of local hot spots due to the inefficient thermal conductivity can cause severe problems. In simulating such problems, a heat transfer model between solid and fluid is necessary.

The performance of the fluid-solid coupled model introduced in Sec. II B 2 is evaluated by conducting a test case in which a heat flow is induced between two planar surface. A rectangular strip of the wall at $y = 0$ is maintained at a temperature of 1.1 [see Fig. 5], whereas the temperature of the rest of the wall and the wall at $y = L_y$ can evolve freely from an initial temperature of 1.0. A 10% maximum difference in temperature is chosen to limit the effects of changes in temperature-dependent properties such as diffusivity, while producing a relatively high signal-to-noise ratio. The latter

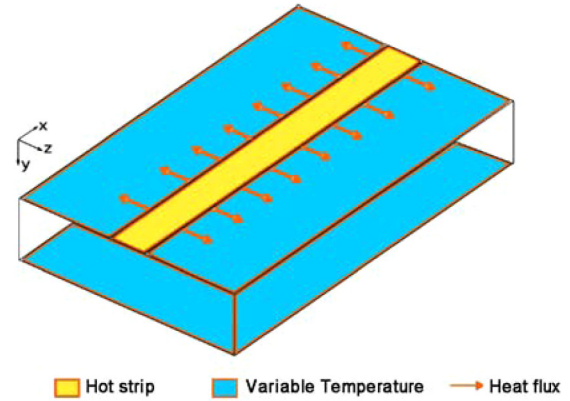


FIG. 5. Schematic of the heated wall. Blue represents the zone with variable temperature, yellow the heated zone and the red arrows denote the direction of heat flow. Domain boundaries in x and z coordinates are periodic.

reduces the needed number of parallel simulations to be run for averaging.

Figure 6 presents the evolution of the local wall surface temperatures with time. The temperatures are averaged in the x direction since the system does not vary along this direction. It can be observed that the conduction through the fluid to the cold wall occurs as expected. Since the temperature of the heated strip is higher than the surroundings, the temperature increase of the fluid starts near the hot strip. The opposite wall slowly increases in temperature due to the thermal diffusion across the gap.

D. Temperature programed desorption and isothermal surface reactions

By now, the foundation for the real-time temperature measurement has been laid. Next, we will demonstrate that the model is suitable for simulating complex surface processes like heterogeneous chemical reactions, which contains three steps: adsorption, reaction, and desorption at catalytic surface sites.

To implement heterogeneous chemical reactions, the Langmuir adsorption-reaction model proposed by Sengar [55] is applied. Details of the model can be found in Appendix.

In the preprocessing step, the catalyst surface is assigned active sites. They can either be distributed uniformly on the surface in a fixed grid pattern, or they can be randomly distributed. In this work, we distribute the active sites randomly onto discretized surface cells. For simplicity, these cells are taken to be the intersections between the solid and the SRD grids. A particle is then adsorbed exactly where it collides with the wall if there are any free active sites in the cell. This way, instead of fixing the locations of each active site, we fix the number of active sites in each cell, which determines the capacity of the cell to hold adsorbed particles. In this way, the catalyst sites can be easily implemented for complex boundaries. It is notable that being coarse-grained particles, the SRD particles represent multiple molecules and similarly the catalyst sites are larger than the size of molecular catalyst sites.

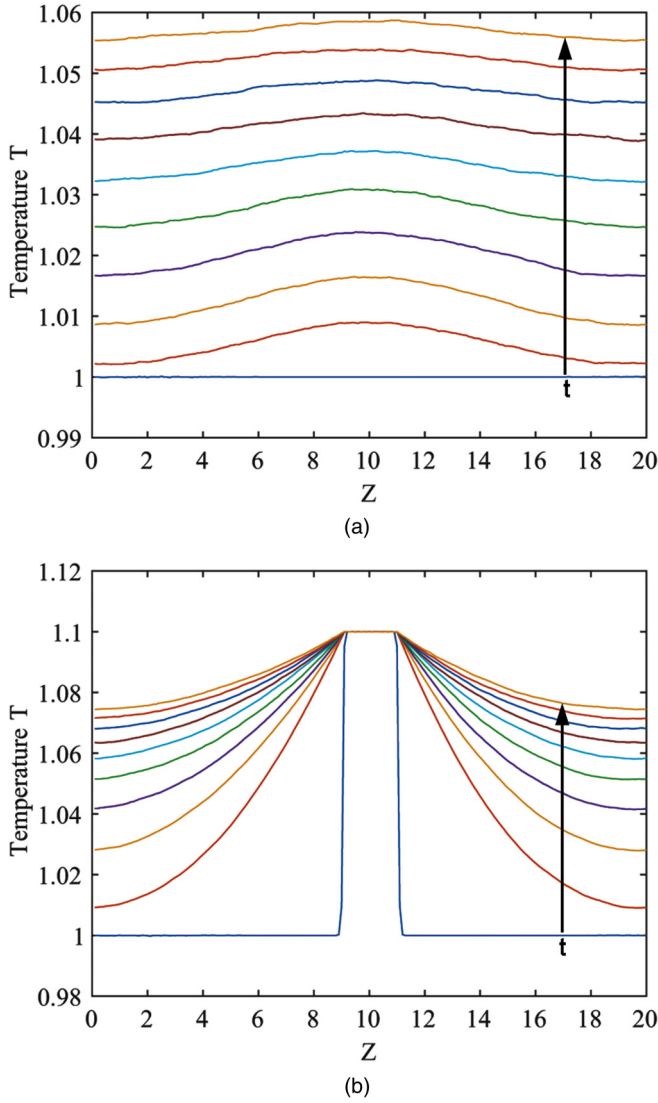


FIG. 6. Temperature evolution of the surface of the unheated (a) and heated (b) wall with time. Subsequent time steps are plotted above each other to illustrate the change in temperature profile with time.

The grid shift procedure needs to be adjusted for the presence of catalytic spheres. Ghost particles are added when the particle density in the SRD cell near a sphere is lower than average number density of the fluid, which is similar to the treatment of the wall of the domain described in Sec. II A. The velocities of these particles are taken from a Gaussian distribution with mean velocity equal to that of the spherical particle (zero for the fixed spheres in this work) and the same temperature as the fluid, as determined in the previous step.

We first perform a virtual experiment of temperature programmed desorption (TPD) from a slowly heated solid spherical catalyst in a bath of SRD particles. The geometry of the test domain is set to $L_x = L_y = L_z = 20$, and the sphere radius is 2 with in total 10 000 catalytic sites on the surface. To load the sphere surface with particles, the adsorption probability prefactor p_a^0 is set to 0.5 and desorption probability prefactor p_d^0 to 0. The simulation is then allowed to run until all N_{cat} catalytic

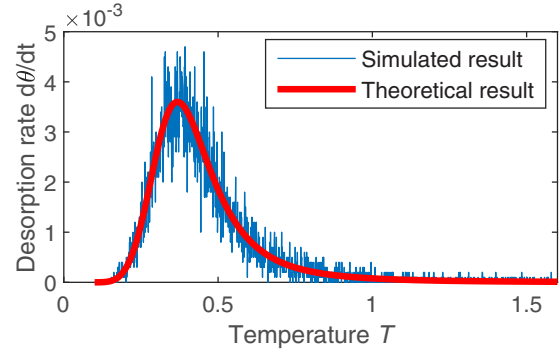


FIG. 7. Desorption rate versus temperature during a TPD virtual experiment where the sphere temperature is increased at a rate of $\beta = 0.001$.

sites per unit area are occupied. The temperature of the solid is then lowered to 0.1 and subsequently slowly increased at a rate of $\beta = 0.001$. The adsorption probability prefactor is set to 0 to avoid re-adsorption. The desorption probability prefactor is set to 1 and the particles are allowed to desorb from the surface as the temperature increases. Δt_s is the streaming time interval as the timestep to update adsorption, reaction, and desorption. The fractional surface coverage of the sphere $\theta = N_a/N_{cat}$, where N_a is the number of adsorbed particles per unit area and N_{cat} is the number of catalytic sites per unit area, will evolve according to the rate of desorption:

$$\frac{d\theta}{dt} = -\theta \frac{p_d^0}{\Delta t_s} e^{-E_d/k_B T}, \quad (16)$$

where E_d is the desorption activation energy, set to $0.5k_B T_0$ in this example. The left-hand side can now be transformed as:

$$\frac{d\theta}{dt} = \frac{d\theta}{dT} \frac{dT}{dt} = \frac{d\theta}{dT} \beta. \quad (17)$$

Therefore, the final expression for the TPD curve can be written as:

$$\frac{d\theta}{dT} = -\frac{p_d^0}{\beta \Delta t_s} \theta e^{-E_d/k_B T}. \quad (18)$$

Equation (18) can be solved numerically to obtain the final solution. This resulting solution, the desorption rate as the change of $\frac{d\theta}{dt}$ with temperature is compared with the simulation results in Fig. 7. It shows that the observed simulation results closely agree with the theoretical prediction, which validates that the desorption model exhibits the expected temperature kinetics. The fluctuations in the simulation results are a consequence of density fluctuations in the SRD fluid.

Next we test the competition among adsorption, reaction, and desorption for an isothermal case (in the next section we will investigate heat effects). At the catalyst surface, an adsorbed reactant particle A reacts to form a product particle B. While both A and B are set to have equal adsorption and desorption probability prefactors as well as activation energies, the desorption probability prefactor is set to a low value of 0.05 to create a rate-limiting step when adsorption probability prefactor is 0.5. This also avoids unwanted hydrodynamic effects due to sudden changes in local density. The probability prefactor for reaction is set to 0.005. The real-time

fractional surface coverage is then measured. As explained in Appendix, the number of A and B particles adsorbing per unit area per unit time can be written as:

$$R_{A,\text{ads}} = C_A \sqrt{\frac{2k_B T}{\pi m_A}} p_a^0 \left(1 - \frac{N_{A,\text{ads}} + N_{B,\text{ads}}}{N_{\text{cat}}}\right) e^{-E_a/k_B T}, \quad (19)$$

$$R_{B,\text{ads}} = C_B \sqrt{\frac{2k_B T}{\pi m_B}} p_a^0 \left(1 - \frac{N_{A,\text{ads}} + N_{B,\text{ads}}}{N_{\text{cat}}}\right) e^{-E_a/k_B T}, \quad (20)$$

where C_A and C_B are the local concentrations (number densities) of A and B particles, respectively, and the term between brackets is the explicit form of $(1 - \theta)$, expressed in terms of the number of adsorbed A and B particles per unit area, $N_{A,\text{ads}}$ and $N_{B,\text{ads}}$, relative to the total number of catalytic sites per unit area, N_{cat} . The number of A and B particles desorbing per unit area per unit time can be written as:

$$R_{A,\text{des}} = N_{A,\text{ads}} \frac{p_d^0}{\Delta t_s} e^{-E_d/k_B T}, \quad (21)$$

$$R_{B,\text{des}} = N_{B,\text{ads}} \frac{p_d^0}{\Delta t_s} e^{-E_d/k_B T}. \quad (22)$$

Finally, the number of A reacting to B per unit area per unit time (assuming an irreversible reaction) can be written as:

$$R_{A,\text{react}} = N_{A,\text{ads}} \frac{p_r^0}{\Delta t_s} e^{-E_r/k_B T}. \quad (23)$$

These equations can now be combined to form a set of differential equations governing the number of A and B on the surface:

$$\frac{dN_{A,\text{ads}}}{dt} = R_{A,\text{ads}} - R_{A,\text{des}} - R_{A,\text{react}}, \quad (24)$$

$$\frac{dN_{B,\text{ads}}}{dt} = R_{B,\text{ads}} - R_{B,\text{des}} + R_{A,\text{react}}. \quad (25)$$

These equations can be solved numerically to give the surface particle numbers with time. In our numerical solution, we assume that the amount of B particles readsorbing onto the surface is negligible, so we effectively set $\gamma_B = 0$.

Figure 8 compares the surface coverage in the numerical solution (blue line) and the SRD model (red dots). It can be seen that the SRD solution closely resembles the numerical solution. The equilibrium total surface coverage can be seen to match the theoretical value, proving that the adsorption and desorption are working as expected.

E. Surface reactions on catalytic particles with heat effects

To further demonstrate the real-time measurement of temperature in heterogeneous catalytic reaction simulations, a model with multiple spherical catalyst particles is tested in a flowing reactant medium. For heterogeneous catalytic reactions in densely packed beds, dead zones in the hydrodynamic fields, which can cause the formation of hot spots in the domain. These hot spots could be catastrophic for the reactor. Simulations can aid in understanding, controlling and preferably preventing the emergence of hot spots.

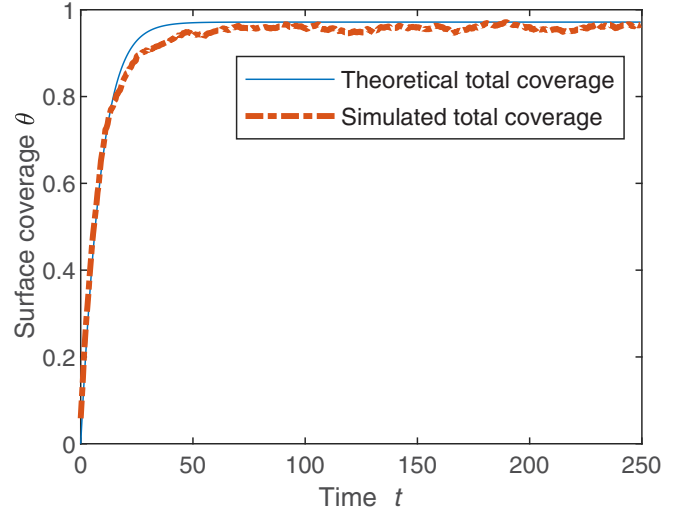


FIG. 8. Time dependence of surface coverage of a catalytic sphere for isothermal adsorption-reaction-desorption.

To replicate the conditions in a densely packed bed reactor where hot spots may form, spherical catalyst particles are configured closely together. Figure 9 shows a schematic of the simulation geometry, which consists of a periodic fluid domain with a thermostatted buffer region at $x = L_x$ spanning a length of 5 units. One, two, or three spheres with a radius of 3 are inserted at $z = L_z/2$, each sufficiently far away from the buffer region. A flow is induced by a uniform forcing term in the x direction. The buffer region returns the outlet fluid to a constant temperature of 1, while also converting B particles back into A, avoiding accumulation of B in the system. L_x is chosen to be 20 (plus buffer region), L_y is 26, and L_z is 16. The coordinates of spheres 1, 2, and 3 are (6,13,13), (12,8,13), and (12,19,13), respectively. In this configuration, when simulating more than one catalytic sphere, the distance between them is set to be slightly larger than 1 SRD cell. This procedure keeps the catalytic spheres close enough to show effects of heat accumulation in the narrow area between the spheres. Individual solid temperature meshes are created for each sphere in the domain. The fluid collision time step

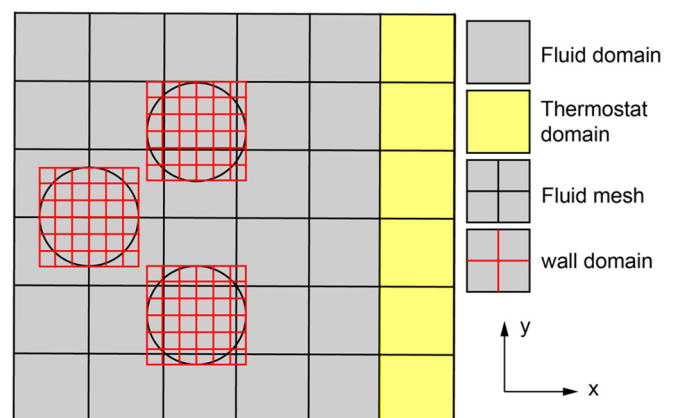


FIG. 9. Schematic overview of the simulation domain (xy plane) for catalytic particles with heat effects.

TABLE III. Parameters for the simulations with surface reactions on catalytic particles (see also Appendix).

Property	Value	Property	Value
Δt_c	0.5	D	0.6249
α	90°	D_T	0.6653
γ	7	ν	0.3503
p_a^0	0.5	E_a	0.5
p_r^0	0.5	E_r	0.5
p_d^0	0.005	E_d	0.5
N_{cat}	100	Δt_s	0.0005
K_S	10	$\rho_S C_S$	100

is set to 0.5 and the local particle density to 7. The flow is driven by a body force $g = 0.001$. The arrangement method of catalytic sites is the same as for the temperature programed desorption case and explained in Sec. III D. The number of catalyst sites per catalytic sphere is 100. The complete simulation settings are listed in Table III. Two hundred fifty simulations were conducted simultaneously to enable real-time temperature measurement.

Figure 10 presents the product density field in the simulation domain. Figures 10(a)–10(c) show the steady-state number density contour on the plane $z = L_z/2$ for the three scenarios. It is clearly visible how in the buffer region the product particles are reset to reactant particles. From the prod-

uct density distribution in Figs. 10(a)–10(c), it can be seen that convection-diffusion of the product from the surface of the catalyst occurs and varies for the three scenarios. The distribution of the product in the domain is biased toward the right due to convection. In the case of multiple spheres, product accumulation happens in the narrow space between spheres, along with the occurrence of hydrodynamic dead spots. Since the reaction is highly desorption limited, the magnitude of the product number density is smaller than that of the reactant.

Figure 10(d) shows the evolution of the total number of B particles in the system with time. The three solid lines represent the three different schemes investigated here. The two dashed lines, named “2 sphere isolated” and “3 sphere isolated,” are constructed by simply scaling the results for “1 sphere” by the number of spheres in the domain. It can be seen that the qualitative behavior remains the same as the number of spheres is scaled up. However, the relative placement of the three spheres can be seen to encourage the entrapment of more product particles in the system, when compared with three independent spheres. This effect can be attributed to the presence of the dead zone between the spheres. The particles that reach these dead zones would have a longer residence time than the rest of the particles, which leads to nonuniformity in the reaction balance within a packed bed.

Each of the scenarios exhibit a quick initial increase, followed by a gradual decrease of the total number of B particles. The initial rise takes place over the first 100 to 200 time steps, which coincides with the time for the development of the velocity field. At these initial times, the effect of convective transport has not reached a significant value, leading to a

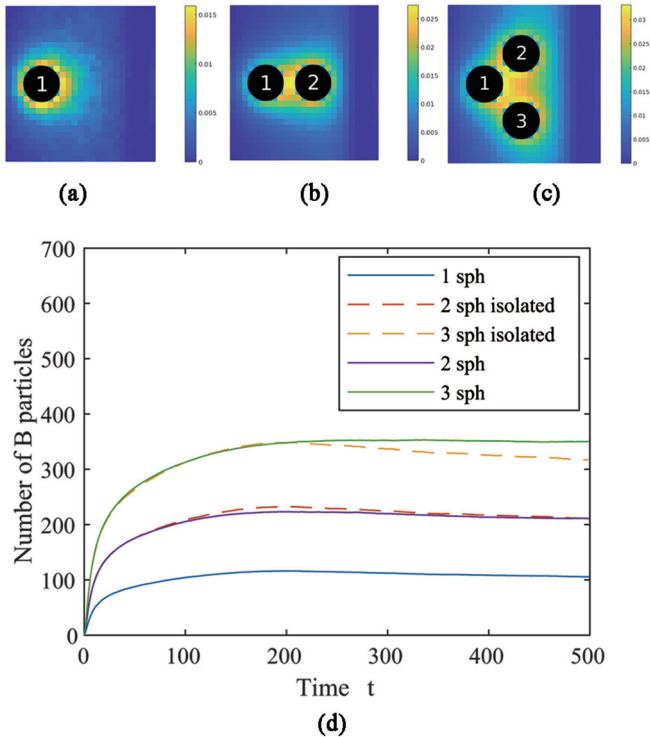


FIG. 10. Mean product number density distribution in a cross section of a domain. Panels (a), (b), and (c) shows results for one, two, and three spheres. Panel (d) compares the total number of B particles in each of the system with the results expected for multiple isolated spheres.

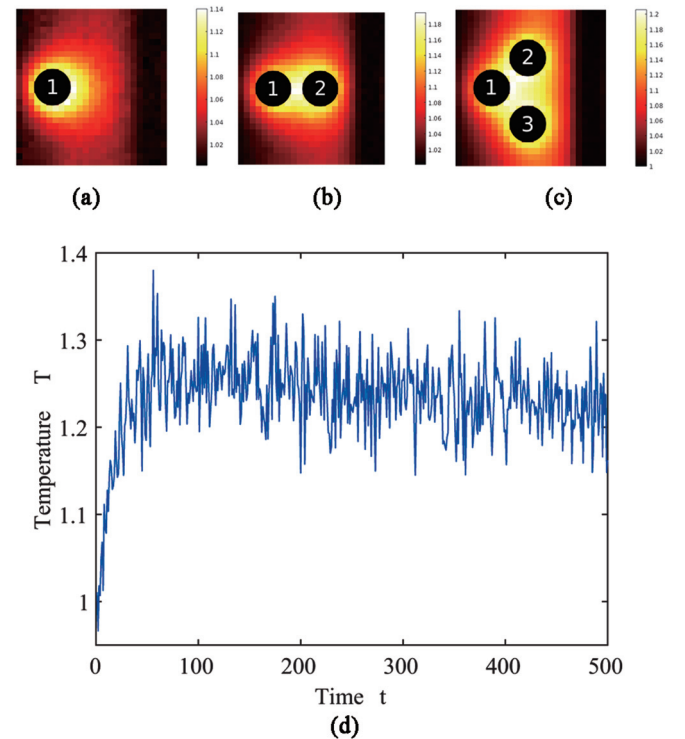


FIG. 11. Steady-state temperature profiles for a cross section of the domain [(a)–(c)]. Panel (d) gives the transient change in local temperature midway between spheres 2 and 3 ($x = 12$ and $y = 14$).

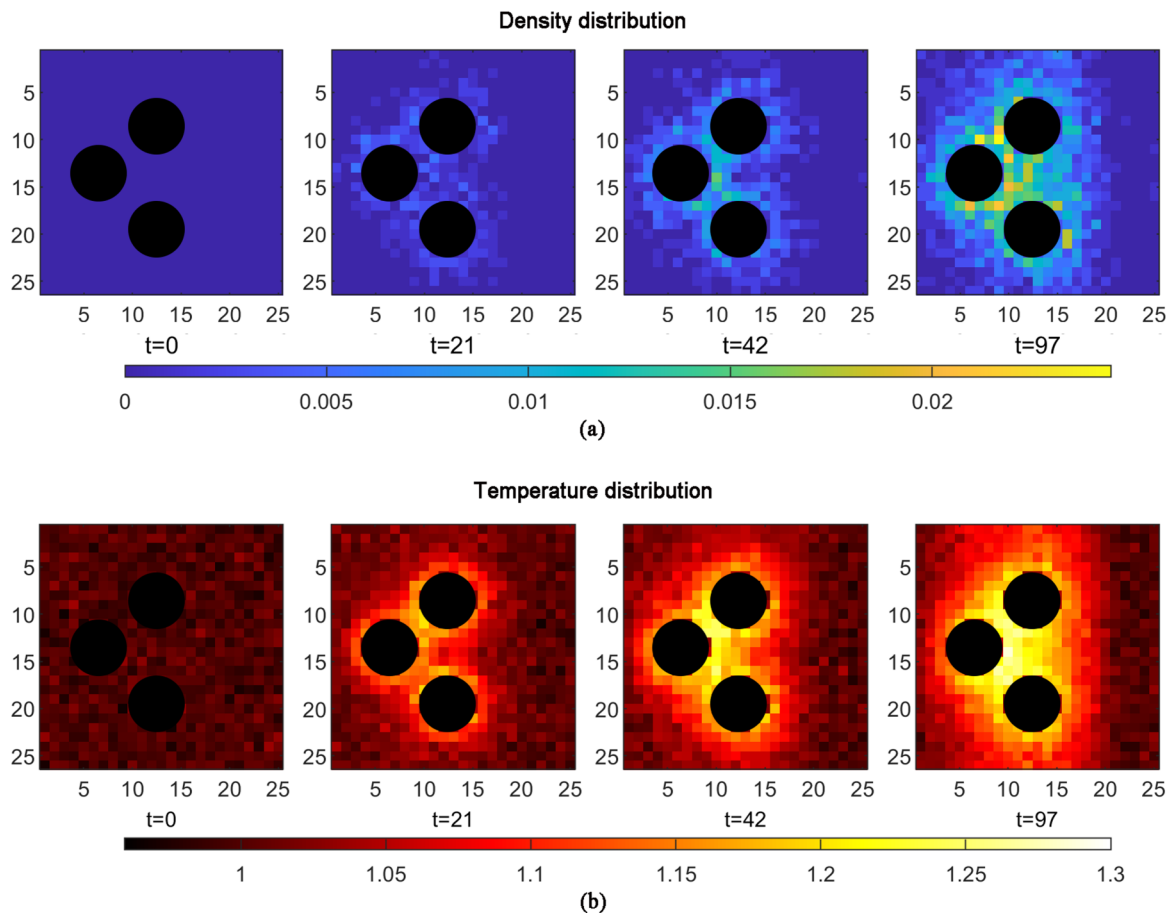


FIG. 12. Variation of local B number density (a) and temperature (b) in cross section of domain with time.

diffusion dominated transport of product. Once the convection picks up, the additional particles away from the spheres are gradually forced to the buffer region and converted back to reactant particles, thereby decreasing the total number of product particles.

Figures 11(a)–11(c) show the steady-state fluid temperature distribution surrounding one, two, and three spheres, respectively, about the plane $z = L_z/2$. Comparing Figs. 10 and 11, it is seen that the temperature contour follows the same qualitative behavior as the product density profile, while the temperature enhancement penetrates further into the fluid than the product does. This is not caused by the diffusion, since both mass and temperature have similar diffusivity in this case (Table II). Instead, this behavior can be attributed to the difference in flux of mass and energy. The energy generated due to the surface reaction is used to increase the local wall temperature. This energy is then transferred to all the particles hitting that location of the wall. In contrast, only one product particle is created per reaction event. This results in a higher flux of heat to the fluid than that of product particles. Figures 11(b) and 11(c) demonstrate the effect of dead spots in the flow profile. The location of the hot spots coincides with that of the density field, which is as expected. Additionally, a slightly lower temperature can be observed in Fig. 11(c) between spheres 1 and 3 compared to 1 and 2. This can once again be attributed to the effect of slightly higher local velocity, refreshing the mixture with relatively cool fluid.

It can also be seen that there is an increase in the magnitude of temperature at the hot spot with increase in number of spheres. This can be attributed to the presence of more sources and a lower heat transfer coefficient to the bulk fluid due to the restricted flow. Figure 11(d) shows the temporal variation of local temperature half-way between spheres 2 and 3 (at $x = L_x/2$). It can be seen to follow the same behavior as the total product particle number density. The initial sharp increase is attributed to the diffusion dominated start-up regime, which is followed by a gradual decrease due to the increasing dominance of convection.

Figure 12 shows the temporal evolution of density and temperature in a cross sectional plane at $z = L_z/2$. At $t = 21$, the density and temperature from each of the spheres has started to affect each other. This gradually leads to the formation of a hot spot, which is seen from the series of figures from $t = 0$ to $t = 97$. This behavior is in line with the observations made in the previous paragraphs. One major observation from this is the difference in timescales for density change and temperature change. In the region between the catalytic spheres, the temperature can reach up to 1.3. This will cause a change in diffusivity, particle density, and viscosity. This influence may induce a natural convection effect. However, with a dominant flow driven by a body force $g = 0.001$, we did not observe obvious natural convection in this simulation.

A final observation from these two graphs is the persistence of noise in the simulations. While 250 simulations are

sufficient to observe the major transient behavior in this case, a considerably higher number of ensemble averages is required to resolve smaller temperature variations.

Based on all the above results, it can be concluded that the newly developed techniques (real-time temperature measurement, surface reactions, wall coupling) are capable of working together in synergy to simulate highly complex reactive systems.

IV. CONCLUSION

In this study, a novel temperature measurement technique for stochastic rotation dynamics was developed, based on kinetic theory. This was then combined with a new form of ensemble averaging to enable real-time measurement of temperature. A proper boundary condition is applied to achieve an accurate constant-temperature no-slip boundary condition. The real-time temperature change due to chemical reaction and heat transfer in fluid phase is achieved and measured.

The newly formulated nonisothermal model was coupled with a temperature conduction equation that can model the formation of complex surface temperature patterns. The coupled method was applied to model surface reactions on catalytic particles. A SRD approach was adopted to model the adsorption, where the SRD particles have a temperature-dependent probability to adsorb onto the limited number of active sites on the surface. The adsorbed particles then react with a rate that depends on the local wall temperature. A wall temperature-dependent desorption model was used to release the products into the fluid. This reaction model was validated against theoretical predictions.

Finally, the combined model was applied to simulate a flow past catalytic spheres. The obtained results highlight the capability of the model to simultaneously solve for the transient evolution of macroscopic properties such as mean velocity, temperature, and density, along with employing accurate wall reaction mechanisms. This expansion of the SRD method to transient and nonisothermal simulations enables the use of this method for the study of highly nonlinear and transient mesoscale problems.

In our future work, we will introduce more complex reaction mechanisms, surface diffusion, and rearrangement based on interactions between adsorbed particles. The method developed in this work can be improved with more accurate treatment of the solid temperature distribution to simulate authentic reactions in porous media and verified with experimental results. This would firmly establish SRD as a promising candidate for simulating reactors with complex geometries and complex surface chemistry.

ACKNOWLEDGMENT

R. Fan thanks the China Scholarship Council (CSC 201707720026) for financial support of this work.

APPENDIX: SIMULATING HETEROGENEOUS REACTIONS

In this work we use the reaction method suggested by Sengar [55]. When a SRD particle bounces on a catalytic surface, the particle will be adsorbed with a certain probability

p_a . In a single time step Δt_s , for adsorbed particles, further probabilities for reaction and desorption can be applied. Based on this, the intrinsic rates in terms of the SRD simulation parameters can be calculated. In the following, to simplify our notation, we will assume that the probabilities per time step are sufficiently small. If large probabilities ($0.1 < p \leq 1$) are used, then the conversion to rates should be done according to $k = -\ln(1 - p)/\Delta t_s$, see Ref. [55] for a detailed explanation.

The kinetic model used by Pooley and Yeomans [45] is extended to derive an expression for the particle collision rate at walls. The collision frequency, i.e., the number of collisions occurring per unit area of the wall per unit time step, is as follows:

$$Z = C_s \langle |v_y| \rangle. \quad (\text{A1})$$

In this expression, C_s is the number density of the particle species of interest and $|v_y|$ is the average velocity of the particles perpendicular to the wall (in this example the wall normal is chosen in the y direction). The expectation value of $|v_y|$ from the Maxwell-Boltzmann distribution is $\sqrt{2k_bT/(\pi m)}$. The adsorption rate, i.e., the number of adsorption events per unit area of catalytic surface per unit time, can therefore be expressed as

$$R_a = C_s \sqrt{\frac{2k_bT}{\pi m}} p_a. \quad (\text{A2})$$

The adsorption probability per collision event p_a depends on local temperature through the activation energy for adsorption E_a , which arises from the Eyring-Polanyi equation in transition state theory, and depends on the fractional catalytic surface coverage $\theta = N_a/N_{\text{cat}}$, where N_a is the number of adsorbed particles per unit area and N_{cat} is the number of adsorption sites per unit area of catalytic surface. In this work, we use a first-order (Langmuir model) dependence on the fraction of empty sites ($1 - \theta$), giving us the following expression for adsorption probability:

$$p_a = p_a^0 (1 - \theta) e^{-E_a/k_bT}, \quad (\text{A3})$$

where p_a^0 is the adsorption probability pre-exponential factor.

If there are $N_{r,a}$ reactant particles adsorbed per unit area catalytic surface (remember that there may also be adsorbed product particles), then we can write the reaction rate, i.e., the number of reactant particles converted to product particles per unit area per unit time, as:

$$R_r = N_{r,a} k_r, \quad (\text{A4})$$

where k_r is the reaction rate which in the simulations is controlled by setting the reaction probability p_r :

$$k_r = \frac{p_r}{\Delta t_s}. \quad (\text{A5})$$

Here p_r is the probability of an adsorbed reactant particle to react to an adsorbed product particle during a single time step Δt_s . This probability can be written in the form of an Arrhenius equation (similarly to the adsorption probability):

$$p_r = p_r^0 e^{-E_r/k_bT}, \quad (\text{A6})$$

where p_r^0 is the reaction probability pre-exponential factor and E_r is the activation energy for the reaction.

A similar approach can be adopted to determine the desorption rate per unit area catalytic surface. For example, focusing

on desorption of product particles, we have:

$$R_d = N_{p,a} k_d, \quad (\text{A7})$$

$$k_d = \frac{p_d}{\Delta t_s}, \quad (\text{A8})$$

$$p_d = p_d^0 e^{-E_d/k_B T}, \quad (\text{A9})$$

where $N_{p,a}$ is the number of adsorbed product particles per unit area, k_d is the desorption rate, p_d is the probability of an adsorbed particle to desorb during a time step Δt_s , E_d is the desorption activation energy, and p_d^0 is the pre-exponential factor. Although in this work they are chosen equal, different desorption activation energies and pre-exponential factors may be chosen for the reactant particles.

-
- [1] J. Mareš, On the development of the temperature concept, *J. Therm. Anal. Calorim.* **60**, 1081 (2000).
- [2] W. T. B. Kelvin, *Mathematical and Physical Papers* (Cambridge University Press, 1890), Vol. 3.
- [3] X. Deng, M. Mao, G. Tu, H. Zhang, and Y. Zhang, High-order and high accurate cfd methods and their applications for complex grid problems, *Commun. Comput. Phys.* **11**, 1081 (2012).
- [4] A. Malevanets and R. Kapral, Mesoscopic model for solvent dynamics, *J. Chem. Phys.* **110**, 8605 (1999).
- [5] G. Gompper, T. Ihle, D. Kroll, and R. Winkler, Multi-particle collision dynamics: A particle-based mesoscale simulation approach to the hydrodynamics of complex fluids, in *Advanced Computer Simulation Approaches for Soft Matter Sciences III* (Springer, Berlin, 2009), pp. 1–87.
- [6] T. Ihle and D. M. Kroll, Stochastic rotation dynamics: A Galilean-invariant mesoscopic model for fluid flow, *Phys. Rev. E* **63**, 020201(R) (2001).
- [7] Z.-Y. Yang, X.-F. Tian, L.-J. Liu, and J.-Z. Chen, Role of hydrodynamic interactions in the deformation of star polymers in poiseuille flow, *Chin. J. Polym. Sci.* **38**, 363 (2020).
- [8] M. Wagner and M. Ripoll, Solvent-induced depletion interactions in multiparticle collision dynamic simulations, *Int. J. Mod. Phys. C* **30**, 1941008 (2019).
- [9] M. Formanek and A. J. Moreno, Single-chain nanoparticles under homogeneous shear flow, *Macromolecules* **52**, 1821 (2019).
- [10] A. Lamura and R. G. Winkler, Tethered semiflexible polymer under large amplitude oscillatory shear, *Polymers* **11**, 737 (2019).
- [11] C.-C. Huang, R. G. Winkler, G. Sutmann, and G. Gompper, Semidilute polymer solutions at equilibrium and under shear flow, *Macromolecules* **43**, 10107 (2010).
- [12] R. Chelakkot, R. G. Winkler, and G. Gompper, Migration of semiflexible polymers in microcapillary flow, *Europhys. Lett.* **91**, 14001 (2010).
- [13] D. Steinhauser, S. Köster, and T. Pfohl, Mobility gradient induces cross-streamline migration of semiflexible polymers, *ACS Macro Lett.* **1**, 541 (2012).
- [14] L. Cannavacciuolo, R. Winkler, and G. Gompper, Mesoscale simulations of polymer dynamics in microchannel flows, *Europhys. Lett.* **83**, 34007 (2008).
- [15] J. T. Padding and A. A. Louis, Hydrodynamic interactions and brownian forces in colloidal suspensions: Coarse-graining over time and length scales, *Phys. Rev. E* **74**, 031402 (2006).
- [16] A. Malevanets and J. Yeomans, Dynamics of short polymer chains in solution, *Europhys. Lett.* **52**, 231 (2000).
- [17] J. T. Padding and A. A. Louis, Hydrodynamic and Brownian fluctuations in sedimenting suspensions, *Phys. Rev. Lett.* **93**, 220601 (2004).
- [18] A. Wysocki, C. P. Royall, R. G. Winkler, G. Gompper, H. Tanaka, A. van Blaaderen, and H. Löwen, Direct observation of hydrodynamic instabilities in a driven non-uniform colloidal dispersion, *Soft Matter* **5**, 1340 (2009).
- [19] I. O. Götze and G. Gompper, Mesoscale simulations of hydrodynamic squirmer interactions, *Phys. Rev. E* **82**, 041921 (2010).
- [20] G. Rückner and R. Kapral, Chemically Powered Nanodimers, *Phys. Rev. Lett.* **98**, 150603 (2007).
- [21] J. L. McWhirter, H. Noguchi, and G. Gompper, Flow-induced clustering and alignment of vesicles and red blood cells in microcapillaries, *Proc. Natl. Acad. Sci. USA* **106**, 6039 (2009).
- [22] M. Yang and M. Ripoll, Simulations of thermophoretic nanoswimmers, *Phys. Rev. E* **84**, 061401 (2011).
- [23] K. Qi, E. Westphal, G. Gompper, and R. G. Winkler, Enhanced Rotational Motion of Spherical Squirmer in Polymer Solutions, *Phys. Rev. Lett.* **124**, 068001 (2020).
- [24] J.-T. Kuhr, F. Rühle, and H. Stark, Collective dynamics in a monolayer of squirmers confined to a boundary by gravity, *Soft Matter* **15**, 5685 (2019).
- [25] A. Zöttl and H. Stark, Simulating squirmers with multiparticle collision dynamics, *Eur. Phys. J. E* **41**, 61 (2018).
- [26] Y.-G. Tao, I. O. Götze, and G. Gompper, Multiparticle collision dynamics modeling of viscoelastic fluids, *J. Chem. Phys.* **128**, 144902 (2008).
- [27] S. Ji, R. Jiang, R. G. Winkler, and G. Gompper, Mesoscale hydrodynamic modeling of a colloid in shear-thinning viscoelastic fluids under shear flow, *J. Chem. Phys.* **135**, 134116 (2011).
- [28] B. Kowalik and R. G. Winkler, Multiparticle collision dynamics simulations of viscoelastic fluids: Shear-thinning gaussian dumbbells, *J. Chem. Phys.* **138**, 104903 (2013).
- [29] S. M. Mousavi, G. Gompper, and R. G. Winkler, Wall entrapment of peritrichous bacteria: a mesoscale hydrodynamics simulation study, *Soft Matter* **16**, 4866 (2020).
- [30] S. C. Kohale and R. Khare, Cross stream chain migration in nanofluidic channels: Effects of chain length, channel height, and chain concentration, *J. Chem. Phys.* **130**, 104904 (2009).
- [31] R. M. Jendrejack, E. T. Dimalanta, D. C. Schwartz, M. D. Graham, and J. J. de Pablo, DNA Dynamics in a Microchannel, *Phys. Rev. Lett.* **91**, 038102 (2003).
- [32] A. P. Minton, How can biochemical reactions within cells differ from those in test tubes? *J. Cell Sci.* **119**, 2863 (2006).
- [33] G. Guigas, C. Kalla, and M. Weiss, The degree of macromolecular crowding in the cytoplasm and nucleoplasm of mammalian cells is conserved, *FEBS Lett.* **581**, 5094 (2007).
- [34] P. Mocchi, P. Derreumaux, F. Sterpone, and S. Melchionna, Mesoscale biosimulations within a unified framework: From proteins to plasmids, *Mol. Simul.* **47**, 101 (2021).

- [35] R. Kapral, Multiparticle collision dynamics: Simulation of complex systems on mesoscales, *Adv. Chem. Phys.* **140**, 89 (2008).
- [36] E. Allahyarov and G. Gompper, Mesoscopic solvent simulations: Multiparticle-collision dynamics of three-dimensional flows, *Phys. Rev. E* **66**, 036702 (2002).
- [37] S. Lepri, G. Ciraolo, P. Di Cintio, J. Gunn, and R. Livi, Kinetic and hydrodynamic regimes in multi-particle-collision dynamics of a one-dimensional fluid with thermal walls, *Phys. Rev. Res.* **3**, 013207 (2021).
- [38] S. Mühlbauer, S. Strobl, M. Coleman, and T. Pöschel, Simulation of catalytic reactions in open-cell foam structures, [arXiv:2010.03904](https://arxiv.org/abs/2010.03904).
- [39] I. V. Pivkin and G. E. Karniadakis, A new method to impose no-slip boundary conditions in dissipative particle dynamics, *J. Comput. Phys.* **207**, 114 (2005).
- [40] D. S. Bolintineanu, J. B. Lechman, S. J. Plimpton, and G. S. Grest, No-slip boundary conditions and forced flow in multi-particle collision dynamics, *Phys. Rev. E* **86**, 066703 (2012).
- [41] A. Lamura, G. Gompper, T. Ihle, and D. Kroll, Multi-particle collision dynamics: Flow around a circular and a square cylinder, *Europhys. Lett.* **56**, 319 (2001).
- [42] T. Ihle and D. M. Kroll, Stochastic rotation dynamics. ii. Transport coefficients, numerics, and long-time tails, *Phys. Rev. E* **67**, 066706 (2003).
- [43] T. Ihle, E. Tüzel, and D. M. Kroll, Equilibrium calculation of transport coefficients for a fluid-particle model, *Phys. Rev. E* **72**, 046707 (2005).
- [44] N. Kikuchi, C. Pooley, J. Ryder, and J. Yeomans, Transport coefficients of a mesoscopic fluid dynamics model, *J. Chem. Phys.* **119**, 6388 (2003).
- [45] C. Pooley and J. Yeomans, Kinetic theory derivation of the transport coefficients of stochastic rotation dynamics, *J. Phys. Chem. B* **109**, 6505 (2005).
- [46] E. Tüzel, M. Strauss, T. Ihle, and D. M. Kroll, Transport coefficients for stochastic rotation dynamics in three dimensions, *Phys. Rev. E* **68**, 036701 (2003).
- [47] E. Tüzel, T. Ihle, and D. M. Kroll, Dynamic correlations in stochastic rotation dynamics, *Phys. Rev. E* **74**, 056702 (2006).
- [48] A. Malevanets and R. Kapral, Solute molecular dynamics in a mesoscale solvent, *J. Chem. Phys.* **112**, 7260 (2000).
- [49] A. Sam, S. K. Kannam, R. Hartkamp, and S. P. Sathian, Water flow in carbon nanotubes: The effect of tube flexibility and thermostat, *J. Chem. Phys.* **146**, 234701 (2017).
- [50] M. P. Howard, A. Z. Panagiotopoulos, and A. Nikoubashman, Efficient mesoscale hydrodynamics: Multiparticle collision dynamics with massively parallel gpu acceleration, *Comput. Phys. Commun.* **230**, 10 (2018).
- [51] J. D. Olarte-Plata and F. Bresme, Microscopic relationship between colloid–colloid interactions and the rheological behaviour of suspensions: A molecular dynamics-stochastic rotation dynamics investigation, *Mol. Phys.* **116**, 2032 (2018).
- [52] Note the subtlety involved through our definition of the energy scale as $k_B T_0$: A unit change in local temperature corresponds to a 0.5 unit change in (kinetic) energy.
- [53] M. Ernst, E. Hauge, and J. Van Leeuwen, Asymptotic Time Behavior of Correlation Functions, *Phys. Rev. Lett.* **25**, 1254 (1970).
- [54] M. Ernst, E. Hauge, and J. Van Leeuwen, Asymptotic time behavior of correlation functions. i. Kinetic terms, *Phys. Rev. A* **4**, 2055 (1971).
- [55] A. Sengar, J. A. M. Kuipers, R. A. Van Santen, and J. T. Padding, Particle-based modeling of heterogeneous chemical kinetics including mass transfer, *Phys. Rev. E* **96**, 022115 (2017).