

Multiscale modeling: foundations, historical milestones, current status, and future prospects

Ravi Radhakrishnan¹

¹University of Pennsylvania

June 9, 2020

Abstract

Research problems in the domains of physical, engineering, biological sciences, often span multiple time and length scales, owing to the complexity of information transfer underlying mechanisms. Multiscale modeling (MSM) and high-performance computing (HPC) have emerged as indispensable tools for tackling such complex problems. We review the foundations, historical developments, and current paradigms in MSM. A paradigm shift in MSM implementations is being fueled by the rapid advances and emerging paradigms in HPC at the dawn of exascale computing. Moreover, amidst the explosion of data science, engineering, and medicine, machine learning (ML) integrated with MSM is poised to enhance the capabilities of standard MSM approaches significantly, particularly in the face of increasing problem complexity. The potential to blend MSM, HPC, and ML presents opportunities for unbound innovation and promises to represent the future of MSM and explainable ML that will likely define the fields in the 21st century.

1. Introduction

Scientific research in the 21st century is characterized by research problems of increasing complexity amidst a data revolution. An ever-growing number of scientific research problems are now focused on systems and processes that are complex not only in terms of their underlying mechanisms and governing principles but also by virtue of the high-dimensional and heterogeneous data worlds that they live in. Modeling, simulation, and high-performance computing, alongside experiments, are indispensable for tackling such problems — numerous success stories have been published across diverse fields. Nonetheless, the unabated increases in complexity and data-intensiveness of modern research problems are now posing three evolving challenges for training a new generation of researchers to have the right tools to navigate the emerging challenges. First, many contemporary problems are now defined over multiple length and time scales (i.e., they are multiscale) and also by multiple distinct, yet intricately coupled, physical, chemical and/or biological processes (they are multiphysics). Solving multiscale-multiphysics problems through multiscale modeling (MSM) methods requires the construction of highly sophisticated algorithms at different scales, the rigorous coupling of the scales, and laborious algorithmic implementation using message passing on parallel high-performance computing (HPC) platforms. Second, the associated increases in data types, data intensiveness, and the types of questions asked, now require more sophisticated approaches for data analysis, including machine learning (ML) techniques, which are becoming indispensable in many applications. Third, MSM and ML approaches have evolved independently, and therefore, the art of combining them is very much an emerging paradigm. This review article describes the convergence of several advances in the scientific literature that has made the field of MSM what it is today and provides a perspective of its future, hoping that it would benefit current and potential researchers navigate and advance the field of MSM.

2. Governing equations for multiphysics modeling

While the considerations above and the motivation to combine MSM and ML can benefit several disci-

plines, it is particularly relevant for chemical, biomolecular, and biological engineers. In our disciplines, the fundamentals (namely, thermodynamics, kinetics, transport, controls) have always emphasized molecular to process length and timescales. These core subjects are rooted in their own foundations, each with its premise, and a set of governing equations are discipline dependent. Statistical mechanics drives much of molecular-scale interactions, quantum mechanics drives catalytic mechanisms, mesoscopic scale relevant to advanced functional materials, energy, or cellular processes are constrained by the laws of transport physics, and foundations of process control and optimizations are rooted in applied mathematics, in particular, in the formal analysis of stability, robustness, evolvability, stochastic effects or noise propagation, and sensitivity analysis [1-4]. In this section, we attempt to provide a unified description of the underlying governing equations in multiphysics modeling. In section 3, we summarize how the foundations and the governing equations have translated into methods and algorithms for multiphysics modeling and simulations. In section 4, we discuss HPC, and in sections 5 and 6 we discuss the current and future prospects of MSM. We end with some conclusions in section 7. We begin by outlining a summary of historical developments of governing equations and foundations for multiphysics modeling in Table 1.

1687	Classical mechanics: the three laws of motion were first compiled by Isaac Newton in his <i>Philosophiae Naturalis Principia Mathematica</i> .
1838	Liouville theorem and equation of classical dynamics: J. Liouville, <i>Journal de Math.</i> , 3, 342 (1838)
1838	Navier Stokes equation: C.L. Navier, <i>Résumé des Leçons Données à l'École des Ponts et Chaussées sur l'Application de la Mécanique à l'Art de Construire</i> .
1865	Maxwell's equations of electrodynamics: James Clerk Maxwell, <i>A Dynamical Theory of the Electromagnetic Field</i> , <i>Philosophical Magazine</i> , 3, 273 (1865).
1870	Boltzmann equation: L. Boltzmann, <i>Weitere Studien über das Wärmegleichgewicht unter Gasmolekülen</i> . <i>Sitzungsberichte der Kaiserlichen Akademie der Wissenschaften in Wien</i> , 59, 274 (1870).
1880	Navier Stokes equation: G.G. Stokes, <i>On the Theories of the Internal Friction of Fluids in Motion, and the Equilibrium of Solids</i> , <i>Philosophical Magazine</i> , 5, 37 (1880).
1908	Langevin equation: Langevin, P. (1908). <i>Sur la théorie du mouvement brownien</i> [On the Theory of Brownian Motion], <i>Comptes Rendus de l'Académie des Sciences et des Beaux-Arts</i> , 146, 530 (1908).
1926	Schrödinger equation: Schrödinger, E. (1926). "An Undulatory Theory of the Mechanics of Atoms and Molecules" (<i>Physical Review</i> , 1, 109 (1926)).
1930	Hartree-Fock method: Slater, J. C. (1930). <i>Note on Hartree's Method</i> . <i>Phys. Rev.</i> 35 (2): 210. doi:10.1103/PhysRev.35.210
1931	Markov process: Kolmogorov, Andrei (1931). <i>Über die analytischen Methoden in der Wahrscheinlichkeitstheorie</i> [On the Analytical Methods in the Theory of Probability], <i>Ann. Inst. Fourier</i> , 1, 107 (1931).
1931	Linear response theory and the fluctuation-dissipation theorem: Onsager's Linear Response Theory: L. Onsager, <i>Physical Review</i> , 37, 405 (1931).
1954	Green-Kubo relations: Green, Melville S. (1954). <i>Markoff Random Processes and the Statistical Mechanics of Time-Reversible Systems</i> , <i>Journal of Chemical Physics</i> , 21, 1099 (1954).
1964	Density functional theory: Hohenberg, Pierre; Walter, Kohn (1964). <i>Inhomogeneous electron gas</i> . <i>Phys. Review</i> , 136, 864 (1964).
1881	Master Equation: van Kampen, N. G. (1981). <i>Stochastic processes in physics and chemistry</i> . North Holland. ISBN 90-01-45333-3
1997	Fluctuation theorems: Jarzynski, C. (1997). <i>Nonequilibrium equality for free energy differences</i> , <i>Phys. Rev. Lett.</i> , 78, 1975 (1997).

Table 1 : Historical milestones of governing equations for multiphysics modeling

Within the foundations of statistical mechanics, any theory based on bottom-up molecular models or top-down phenomenological models is developed with the notion of microstates accessible by a system. The dynamics of the system at this level can be described based on transitions between microstates. A microstate defines the complete set of configurations accessible to the system (e.g., positions and momenta of all the particles/molecules of the system). For molecular systems obeying laws of classical dynamics (Newton's laws), the microstate of the system with a given set of positions and momenta at a given time t only depends on the microstate at the immediately preceding time step. This memory-less feature is a hallmark of a Markov process, and all Markov processes obey the master equation [5]. Note that the Markov process is very general, and the classical dynamics is just a particular case. The probability of access to a microstate defined by a given value of the microstate variables y is denoted by $P(y,t)$, which is time-dependent for a general dynamical process at nonequilibrium. A set of probability balance equations governs Markov processes (under certain assumptions), collectively referred to as the master equation given by:

$$\frac{dP(y,t)}{dt} = \sum_{y'} [w(y'|y)P(y',t) - w(y|y')P(y,t)]. \quad (\text{Eq. 1})$$

Here, y and y' denote different microstates and $w(y|y')$ is the transition probability (which is a rate of transition in units of a frequency) from state y' to state y .

The Liouville equation in classical dynamics is a particular case of the continuous version of the master equa-

tion where the microstates are enumerated by the positions and momenta of each particle [6, 7]. Newtonian dynamics obeys the Liouville equation and the parent master equation, which is easy to see by recognizing that the elements of the transition probabilities under Newtonian dynamics are delta functions and therefore Newtonian dynamics trivially satisfies the Markov property. Similarly, the Schrödinger equation, which governs the dynamics of quantum systems is consistent with the quantum master equation [8]. Therefore, the laws of classical and quantum dynamics are both slaves to the master equation (Eq. 1). Neither the Schrödinger equation nor Newton's equations can predict the interactions between systems (such as atoms and molecules), for which one needs to invoke Maxwell's equations to determine the nature of the potential energy functions [9].

Macroscopic conservation equations can be derived by taking the appropriate moment in (Eq. 1):

$$\langle y \rangle_t = \int dy dy' (y' - y) w(y' | y) P(y, t). \quad (\text{Eq. 2})$$

Here, $\langle y \rangle$ represents the average of y over all states, weighted by the probability of accessing each state. Indeed, a particular case of the master equation is the Boltzmann equation [10], where the microstates defined in terms of the positions and momenta of all particles are reduced to a one-particle (particle j) distribution by integrating over the remaining $n-1$ particles. Here, the operator for the total derivative d/dt is expressed as the operator for the partial derivative $\partial/\partial t$ plus the convection term $u \cdot \nabla$, where u is the velocity. The moments of the Boltzmann equation were derived by Enskog for a general function y_i (here i indexes the particle) [10]. Substituting y as m_i , the mass of particle i yields the continuity equation, as $m_i v_i$, the momentum of particle i yields the momentum components of the Navier-Stokes equation, and as $1/2 m_i v_i^2$, the kinetic energy of the particle, yields the energy equation, which together represents conservation equations that are the pillars of continuum hydrodynamics. Similarly, the rate equations for describing the evolution of species concentrations of chemical reactions can be obtained by computing the moment of the number of molecules using an analogous version of (Eq. 2) known as the chemical master equation [5].

2.1 Thermal and Brownian effects

One of the main attributes of statistical mechanics of equilibrium and nonequilibrium systems that differentiate it from traditional hydrodynamics is that the kinematics and thermal effects have to be treated with equal importance. It is worth noting that while the thermal effects and fluctuations are described within the scope of the master equation (Eq. 1), by taking the moment (average) to derive the conservation law (Eq. 2), often the thermal effects are averaged out to produce only a mean-field equation. Indeed, the continuity, momentum (Navier-Stokes), and energy equations cannot accommodate thermal fluctuations that are inherent in Brownian motion even though such effects are fully accommodated at the level of the parent master equation. Therefore, nanoscale fluid dynamics (NFD) must be approached differently than traditional hydrodynamics.

One approach starts with the mean-field conservation equation, such as the Boltzmann equation, and adds the thermal fluctuations as a random forcing term, which results in the Boltzmann-Langevin equation derived by Bixon and Zwanzig [11]. This approach amounts to random fluctuating terms being added as random stress terms to the Navier-Stokes equations. The above procedure, referred to as the fluctuating hydrodynamics (FHD) approach, was first proposed by Landau and Lifshitz [12]. In the FHD formulation, the fluid domain satisfies:

$$\nabla \cdot \mathbf{u} = 0$$

$$\rho \frac{d\mathbf{u}}{dt} = \rho [\nabla \cdot \boldsymbol{\tau} + \mathbf{v} \cdot \nabla \mathbf{u}] = \nabla \cdot \boldsymbol{\sigma}, \quad (\text{Eq. 3})$$

where, \mathbf{u} and ρ are the velocity and density of the fluid respectively, and $\boldsymbol{\sigma}$ is the stress tensor given by, $\boldsymbol{\sigma} = p\mathbf{J} + \mu [\nabla \mathbf{u} + (\nabla \mathbf{u})^T] + \mathbf{S}$. Here, p is the pressure, \mathbf{J} is the identity tensor, and μ is the dynamic viscosity. The random stress tensor \mathbf{S} is assumed to be a Gaussian white noise that satisfies:

$$\langle S_{ij}(x, t) \rangle = 0$$

$$\langle S_{ij}(x, t) S_{lm}(x', t') \rangle = 2k_B T \mu (\delta_{il} \delta_{km} + \delta_{im} \delta_{kl}) \delta(x - x') \delta(t - t'), \quad (\text{Eq. 4})$$

where, $\langle \cdot \rangle$ denotes an ensemble average, $k_B T$ is the Boltzmann constant, T is the absolute temperature, and δ_{ij} is the Kronecker delta. The Dirac delta functions $\delta(x-x')$ and $\delta(t-t')$ denote that the components of the random stress tensor are spatially and temporally uncorrelated. The mean and variance of the random stress tensor of the fluid are chosen to be consistent with the fluctuation-dissipation theorem [13]. By including this stochastic stress tensor due to the thermal fluctuations in the governing equations, the macroscopic hydrodynamic theory is generalized to include the relevant physics of the mesoscopic scales ranging from tens of nanometers to a few microns.

An alternative approach to NFD (and one that is different from FHD) is to start with a form of the master equation referred to as the Fokker-Planck equation. Formally, the Fokker-Planck equation is derived from the master equation by expanding $w(y' | y) P(y,t)$ as a Taylor series in powers of $r=y'-y$. The infinite series is referred to as the Kramers-Moyal expansion, while the series truncated up to the second derivative term is known as the Fokker-Planck or the diffusion equation, which is given by [5]:

$$\frac{\partial P(y,t)}{\partial t} = - \frac{\partial}{\partial y} [a_1(y)P] + \frac{\partial^2}{\partial y^2} [a_2(y)P]. \quad (\text{Eq. 5})$$

Here, $a_n(y) = \int r^n w(r) dr$. The solution to the Fokker-Planck equation yields the probability distributions of particles which contain the information on Brownian effects. At equilibrium (i.e., when all the time-dependence vanishes), the solution can be required to conform to the solutions from equilibrium statistical mechanics. This approach leads to a class of identities for transport coefficients, including the famous Stokes-Einstein diffusivity for particles undergoing Brownian motion to be discussed later in this article. Furthermore, there is a one-to-one correspondence between the Fokker-Planck equation and a stochastic differential equation (SDE) that describes the trajectory of a Brownian particle. The generalized Fokker-Planck equation is written in terms of a generalized order parameter (or sometimes referred to as a collective variable) S , given by:

$$\frac{\partial P(S,t)}{\partial t} = \frac{D}{k_B T} \frac{\partial}{\partial S} [P(S,t) \frac{\partial F(S)}{\partial S}] + D \frac{\partial^2 P(S,t)}{\partial S^2}, \quad (\text{Eq. 6})$$

where, $F(S)$ is the free energy density (also referred to as the Landau free energy) along S [14], D is the diffusion coefficient along S , which is also related to the a_n 's of the original Fokker-Planck equation, i.e., $a_2=2D$. The quantity $k_B T$, which has the units of energy, is called the Boltzmann factor and serves as a scale factor for normalizing energy values in NFD. Corresponding to every generalized Fokker-Planck equation (Eq. 6), there exists a SDE given by:

$$\frac{dS}{dt} = - \frac{D}{k_B T} \frac{\partial F(S)}{\partial S} + (2D)^{1/2} \xi(t), \quad (\text{Eq. 7})$$

where, $\xi(t)$ represents a unit-normalized white noise process. The SDE encodes for the Brownian dynamics (BD) of the particle in the limit of zero inertia. When the inertia of the particle is added, the corresponding equation is often referred to as the Langevin equation [13]. In summary, Brownian or thermal effects are described in the hydrodynamics framework, either using the FHD or the BD/Langevin equation approach.

2.2 Linear response

Thus far, our discussion has not distinguished between a single system or an interacting system. A general framework for describing its dynamics as well as the equilibrium properties of interacting systems approaching equilibrium can be understood in light of the linear response theory, which is the foundation of nonequilibrium thermodynamics. A system at equilibrium evolving under a Hamiltonian H experiences a perturbation $\Delta H=fA$, where f is the field variable (such as an external force), and A is the extensive variable (such as the displacement) that is conjugate to the field. The perturbation throws the system into a nonequilibrium state, and when the field is switched off, the system relaxes back to equilibrium in accordance with the regression process described by Onsager [13]:

$$\Delta A(t) = \frac{f}{k_B T} \langle \Delta A(0) \Delta A(t) \rangle \quad (\text{Eq. 8})$$

where, $\Delta A(t) = A(t) - \langle A \rangle$. The above identity holds under linear response, when ΔH is small, or equivalently when $\Delta A(t, \lambda f) = \lambda \Delta A(t, f)$. The most general form to relate the response A to the field f under the linear response is given by: $\Delta A(t) = \int \zeta(t-t') f(t') dt'$. Here, we have further assumed that physical processes are

stationary in the sense that they do not depend on the absolute time, but only the time elapsed, i.e., $\zeta(t, t') = \zeta(t-t')$. One can use the linear-response relationship to derive an equation for the dynamics of a system interacting with a thermal reservoir of fluid (also called a thermal bath). For example, the dynamics of the particle (in one-dimension along the x -coordinate for simplicity of illustration is given by $m d^2U/dt^2 = -dV(x)/dx + f$, U is the particle velocity, where $V(x)$ is the potential energy function, and f is an external driving force including random Brownian forces from the solvent degrees of freedom. The thermal bath will experience forces f_r in the absence of the particle, and when the particle is introduced, the perturbation will change the bath forces to f . This change $f-f_r$ can be described under linear response as: $\Delta f(t) = f - f_r = \int_{\zeta_b} \zeta_b(t-t') x(t') dt'$. Using this relationship, and by performing integration by parts, the particle dynamics may be written as:

$$m d^2U/dt^2 = -dV(x)/dx + f_r - \int_{\zeta_b} \zeta_b(t-t') U(t') dt' \quad (\text{Eq. 9})$$

Here the subscript b stands for bath, $\zeta_b(t) = -d\xi_b/dt$, and f_r is the random force from the bath that is memoryless. This form of the equation for the dynamics of the interacting system is referred to as the generalized Langevin equation, and it accounts for the memory/history forces. We note that while the parent equation (i.e., the master equation) is Markovian, the memory emerges as we coarse-grain the timescales to represent the system-bath interactions and is a consequence of the 2nd law of thermodynamics. One can recover the Langevin equation from the GLE by assuming that the memory function in the integral of (Eq. 9) is a Dirac delta function. The strength of the random force that drives the fluctuations in the velocity of a particle (as noted in the above example) is fundamentally related to the coefficient representing the dissipation or friction present in the surrounding viscous fluid. This is the fluctuation-dissipation theorem [15]. The friction coefficient, ζ_b , associated is time-dependent and not given by the constant value (given by the Stokes formula or a drag coefficient). In any description of system dynamics, and therefore, the mean and the variance of observables under the thermal fluctuations have to be chosen to be consistent with the fluctuation-dissipation theorem. In order to achieve thermal equilibrium, the correlations between the state variables should be such that there is an energy balance between the thermal forcing and the dissipation of the system as required by the fluctuation-dissipation theorem [15]. Finally, we note that the fluctuation theorems of Crooks and the Jarzynski relationships for relating equilibrium free energies to nonequilibrium work can be derived from ratios of the probabilities of the forward and backward paths of a Markov process [16].

2.3 Equilibrium and transport properties

According to equilibrium statistical mechanics, in a uniform temperature fluid, the molecular velocities will be Maxwellian, and the energy components related to the various degrees of freedom will satisfy the equipartition principle. Thus, the equilibrium probability density function (PDF) of each of the cartesian components of the particle in the above example U_i , will follow the Maxwell-Boltzmann (MB) distribution. Another important application of the Onsager regression relationship (Eq. 6) is the emergence of a class of relationships that relate transport properties to correlation functions known as the Green-Kubo relationships [13, 17]. These relationships are also a consequence of the fluctuation-dissipation theorem. Thus,

$$\gamma = (1/d) \int_0^\infty dt \langle A(0) \cdot A(t) \rangle. \quad (\text{Eq. 10})$$

Here, γ is the transport coefficient of interest, t is time, d is the dimensionality, A is the current that drives it. The integrand of (Eq. 10) is the autocorrelation function (ACF) of quantity A . One can calculate the transport coefficients such as diffusion D , shear viscosity η_s , and thermal conductivity k using the Green-Kubo formula.

3. Algorithms for multiphysics models in scientific computing

Numerical analysis in applied mathematics and computational chemistry have laid the foundations of much of the algorithms for numerical solving the governing equations in multiphysics modeling. A sketch of the historical developments in the field of numerical analysis that has laid the foundations for much of scientific computing is provided in Table 2. Summaries of algorithms (methods) for multiphysics modeling at different

resolutions (length or timescales) are provided in this section (see also Figure 1). Note that several reviews in the literature summarize these methods to varying degrees of detail, see, for example, [18] and references therein.

1941	Numerical solvers for partial differential equations (PDE): Hrennikoff, Alexander (1941). Solution of problems
1947	Numerical linear algebra: John von Neumann and Herman Goldstine, Numerical Inverting of Matrices of High
1953	Monte Carlo method: Metropolis, N.; Rosenbluth, A.W.; Rosenbluth, M.N.; Teller, A.H.; Teller, E. (1953). Equ
1960-1970	The finite element method: FEM 60s and 70s: Strang, Gilbert; Fix, George (1973). An Analysis of The Finite
1970	Electronic structure methods in computational chemistry: Gaussian is a general-purpose computational chemis
1974-1977	The first molecular dynamics simulation of a realistic system; the first protein simulations. Stillinger, F. H. and
1970s	Development of linear algebra libraries: linear algebra package (LAPACK) https://en.wikipedia.org/wiki/LAPACK
1980-2010	Development of parallel algorithms for linear algebra, Fourier transforms, N-body problems, graph theory [http
2010-2020	Parallel algorithms for machine learning [https://cvw.cac.cornell.edu/APC/]

Table 2 : Historical milestones in numerical analysis and simulations

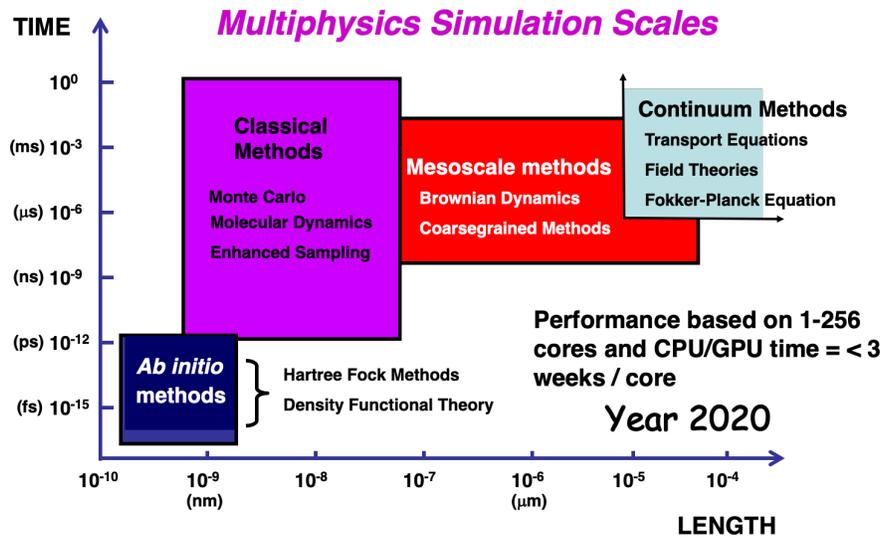


Figure 1 : Multiphysics simulations and capabilities of current systems in high-performance computing platforms available to US researchers such as the extreme science and engineering discovery environment (XSEDE; xsede.org).

3.1 Ab initio electronic structure methods

Foundations of electronic structure methods are based on the variational theorem in quantum mechanics that states that the exact wave function of the ground state of a given Hamiltonian alone is the solution of the variational minimization of the expectation value of the Hamiltonian: minimize $\langle \Psi | H | \Psi \rangle$ subject to $\langle \Psi | \Psi \rangle = 1$ yields $H | \Psi \rangle = E | \Psi \rangle$. In this manner, the variational theorem solves the time-independent Schrodinger equation, which conforms to the quantum master equation, as noted above. As a practical implementation, one can arrive at very close approximations to the exact groundstate solution by expanding the wavefunction in terms of finite basis sets: $|\Psi\rangle = \sum_i c_i |\Phi_i\rangle$. Lynchpin methods that enable the implementation of the variational calculation for many-electron systems, which are further subject to the constraints of Pauli's exclusion principle, are Hartree-Fock methods and methods based on electronic density functional theory.

3.1.1 The Hartree-Fock (HF) approximation : HF corresponds to the conventional single-electron picture of the electronic structure where the distribution of the N electrons is given simply by the product of one-

electron distributions. Hartree-Fock theory, by assuming a determinant form for the wavefunction, imposes the property of antisymmetry; nevertheless, the form neglects correlation between electrons. The electrons are subject to an *average* non-local potential arising from the other electrons, which can lead to an inadequate description of the electronic structure. Although qualitatively correct in many materials and compounds, Hartree-Fock theory can be insufficient to make accurate quantitative predictions. These predictions can be improved using higher-order perturbation theory-based methods [19].

3.1.2 Density functional theory (DFT): DFT is a formally exact theory [20]. It is distinct from quantum chemical methods in that it is a non-interacting theory and does not yield a correlated N-body wavefunction. DFT has come to prominence over the last decade as a method capable of very accurate results at a low computational cost. In practice, approximations are required to implement the theory and the accuracy is context-dependent. The Hohenberg-Kohn theorem states that if N interacting electrons move in a potential external $V_{\text{ext}}(\mathbf{r})$, the groundstate electron density $n_0(\mathbf{r})$ minimizes the energy functional $E[n(\mathbf{r})]$. The practical utility of DFT is in constructing the energy functional by augmenting a free electron gas reference energy functional (which is precisely known) with a parameterized form of energy terms that account for exchange and electron correlation (determined based on more accurate techniques such as quantum Monte Carlo or QMC methods, see below). Variational techniques similar to those utilized in HF methods can then be employed to obtain the ground state solution in DFT.

Softwares for quantum chemical calculations are available under open source or commercial licenses that make it easy to model molecular systems using electronic structure methods:

(https://en.wikipedia.org/wiki/List_of_quantum_chemistry_and_solid-state_physics_software). They have been the driving force to parametrize the force fields of classical simulations such as those in (Eq. 11) below.

3.2 Molecular dynamics

Molecular dynamics (MD) simulation techniques directly solving Newton’s equations of motion are commonly used to model systems of biomolecules and biomaterials because they can track individual atoms and, therefore, answer questions to specific material properties [21, 22]. In MD simulations, the starting point is defining the initial coordinates and initial velocities of the atoms characterizing the model system, for example, the desired biomolecule plus the biologically relevant environment, i.e., water molecules or other solvent and/or membranes. The coordinates of the desired biomolecule can usually be found as structural data (X-ray or NMR) deposited into the protein data bank (PDB) [23] (www.pdb.org); otherwise, it is possible to derive initial geometry and coordinate data from model building techniques, including homology methods. This step also typically includes the placement and positioning of the environment of the molecules (solvation, ionic strength, etc.). The initial velocities are typically derived from the Maxwell-Boltzmann distributions at the desired temperature of the simulation. The potential of interactions of each of the atoms is calculated using a force field function, which parameterizes the non-bonded and bonded interaction terms of each atom depending on its constituent atom connectivity: bond terms, angle terms, dihedral terms, improper dihedral terms, non-bonded Lennard-Jones terms, and electrostatic terms. The potential interactions are summed across all the atoms contained in the system, to compute an overall potential energy:

$$U(R) = \sum_{\text{bonds}} K_b (b - b_0)^2 + \sum_{\text{angles}} K_\theta (\theta - \theta_0)^2 + \sum_{\delta\eta\epsilon\delta\rho\alpha\lambda\zeta} K_\chi (1 + \cos(\eta\chi - \delta)) + \sum_{\psi\mu\rho\sigma\pi\epsilon\rho\zeta} K_\phi (\phi - \phi_0)^2 + \sum_{\nu\sigma\nu\beta\sigma\nu\delta\epsilon\delta} \dots \quad (11)$$

Taking the derivative of the potential energy function yields the force, and from Newton’s second law, this is equal to mass times acceleration. Although the process seems simple, the derivative function results in a set of 3N-coupled 2nd order ordinary differential equations that must be solved numerically. The solution consists of a numerical recipe to advance the positions and the velocities by one timestep. This process is repeated over and over again to generate MD trajectories of constant energy. Constant temperature dynamics are derived by coupling the system to a thermostat using well-established formulations such as the Langevin dynamics or the Nose-Hoover methodologies [24]. Ap-

plication of MD simulations to biomolecules is facilitated by several popular choices of force fields such as CHARMM27 [25] (www.charmm.org), AMBER [26] (www.ambermd.org), and GROMOS [27] (www.gromacs.org), as well as dynamic simulations packages and visualization/analysis tools such as NAMD [28] (www.ks.uiuc.edu/Research/namd/) and VMD [29] (www.ks.uiuc.edu/Research/vmd/). MD simulations for commonly modeled molecules such as proteins, nucleic acids, and carbohydrates that have well-established force fields can be performed directly using a favorite software package such as LAMMPS, GRO-MACS or HOOMD-blue (<http://glotzerlab.engin.umich.edu/hoomd-blue>).

3.3 Monte Carlo (MC), quantum Monte Carlo (QMC), and kinetic Monte Carlo (KMC) methods

In the limit of steady-state, the master equation in (Eq. 1) can be written in matrix form as $WP = P$ or in the familiar Einstein's convention of $w_{ij}P^e_j = P^e_i$, where the summation over the repeated index is implicitly assumed. It is important to recognize is that W is the entire matrix and w_{ij} is the ij^{th} element of the matrix. Note that w_{ij} is the transition probability of migrating from microstate j to i , consistent with the definition of w in (Eq. 1). Similarly, P is the entire vector of probabilities of each microstate, and the i^{th} element of P is P_i , the probability to access microstate i . Note that here, P^e_i is the equilibrium distribution. More generally, if we start with a non-equilibrium state $P(1)$, here, (1) is the initial time and the system transitions to later times and is tracked by (2); (3), etc., then: $WP(1) = P(2); WP(2) = P(3); WP(3) = P(4); \dots; WP(n) = P(n+1)$, and as n becomes large, $P(n) = P(n+1) = P^e$. In a Monte Carlo simulation, we simulate the system by sampling accessible microstates according to their equilibrium distribution P^e , e.g., as given by the Boltzmann distribution or the appropriate equivalent distribution in different ensembles (for thermodynamic systems at equilibrium). To achieve this task, we need to choose a W or all w_{ij} that make up the W , such that $P^e_i = \exp(-E_i/k_B T) / [\sum_j \exp(-E_j/k_B T)]$ is satisfied. Metropolis et al. [30] recognized that this could be achieved by choosing w_{ij} that satisfy equation $w_{ij}P^e_j = P^e_i$ by imposing a stronger criterion, namely, $P^e_m w_{nm} = P^e_n w_{mn}$, which leads to the Metropolis Monte Carlo method for sampling microstates of a classical system.

Quantum Monte Carlo (QMC) techniques provide a direct and potentially efficient means for solving the many-body Schrödinger equation of quantum mechanics [31]. The simplest quantum Monte Carlo technique, variational Monte Carlo (VMC), is based on a direct application of Monte Carlo integration to calculate multi-dimensional integrals of expectation values such as the total energy. Monte Carlo methods are statistical, and a key result is that the value of integrals computed using Monte Carlo converges faster than by using conventional methods of numerical quadrature, once the problem involves more than a few dimensions. Therefore, statistical methods provide a practical means of solving the full many-body Schrödinger equation by direct integration, making only limited and well-controlled approximations.

The kinetic Monte Carlo (KMC) method is a Monte Carlo method computer simulation intended to simulate the time evolution of processes that occur with known transition rates among states (such as chemical reactions or diffusion transport). It is essential to understand that these rates are inputs to the KMC algorithm; the method itself cannot predict them. The KMC method is essentially the same as the dynamic Monte Carlo method and the Gillespie algorithm [32]. From a mathematical standpoint, solving the master equation for such systems is impossible owing to the combinatorially large number of accessible microstates, even considering that the limited number of accessible states renders the transition probability matrix sparse. The Gillespie algorithm provides an ingenious way out of this issue. The practical idea behind KMC is not to attempt to deal with the entire matrix, but instead to generate stochastic trajectories that propagate the system from state to state (i.e., a Markovian sequence of discrete hops to random states happening at random times). From this, the correct time evolution of the probabilities $P_i(t)$ is then obtained by ensemble averaging over these trajectories. The KMC algorithm does so by selecting elementary processes according to their probabilities to fire, followed by an updating of the time.

3.4 Particle-based mesoscopic models

Earlier, we outlined the connection between the Boltzmann equation (a particular case of the master equation) and the continuum transport equations. However, at the nano to mesoscopic lengthscales, neither the

molecular description using molecular dynamics nor a continuum description based on the Navier-Stokes equation are optimal to study nanofluid flows. The number of atoms is too large for MD to be computationally tractable. The microscopic-level details, including thermal fluctuations, play an essential role in demonstrating the dynamic behavior, an effect which is not readily captured in continuum transport equations. Development of particle-based mesoscale simulation methods overcomes these difficulties, and the most common coarse-grained models used to simulate the nanofluid flows are Brownian dynamics (BD) and multi-particle collision dynamics (MPCD) methods. The general approach used in all these methods is to average out relatively insignificant microscopic details in order to obtain reasonable computational efficiency while preserving the essential microscopic-level details.

3.4.1 Brownian dynamics (BD) simulations: The physical system of nanofluids contains relatively small solvent molecules and relatively larger nanoparticles, which move much more slowly due to their larger size. A broad range of time scales, from short time steps for the fast motion to very long runs for the evolution of the slower mode, needs to be accommodated by any simulation method as applied to nanofluids, making the process time-consuming. However, in the BD simulation technique, explicit solvent molecules are replaced by a stochastic force, and the hydrodynamic forces mediated by them are accounted for through a hydrodynamic interaction (HI) kernel. The BD equation thus replaces Newton's equations of motion in the absence of inertia:

$$\mathbf{r}_i = \mathbf{r}_i^0 + \sum_j \frac{D_{ij}^0 \mathbf{F}_j^0}{k_B T} \Delta t + \mathbf{R}_i(\Delta t),$$

(Eq. 12)

where, the superscript 0 denotes the value of the variable at the beginning of the time step, \mathbf{r}_i is the position of the i^{th} nanoparticle, D_{ij} is the diffusion tensor, and \mathbf{F}_j refers to the force acting on the j^{th} particle. The displacement \mathbf{R}_i is the unconstrained Brownian displacement with a white noise having an average value of zero and a covariance of $2D_{ij}^0 \delta(t)$. The Rotne-Prager-Yamakawa hydrodynamic mobility tensor [33, 34] is a commonly employed diffusion tensor to approximate the hydrodynamic interactions mediated by the fluid. The trajectories and interactions between the coarse-grained molecules are calculated using the stochastic differential equation (Eq. 12), which is integrated forward in time, allowing for the study of the temporal evolution and the dynamics of complex fluids. Stokesian dynamics also represent a class of methods under this paradigm [35].

3.4.2 Multi-particle collision dynamics (MPCD): Multi-particle collision dynamics (MPCD) is an algorithm that can model both hydrodynamic interactions and Brownian motion with relatively low computational costs [36]. The algorithm consists of discrete streaming and collision steps at fixed discrete time intervals that have been shown to yield the correct long-time hydrodynamics. The effects of Brownian motion and hydrodynamic interactions are incorporated into the simulation through the collision step. The solvent is characterized by a large number N of point-like particles with a given mass m that move in space with a continuous distribution of velocities. The positions of the solvent particles $\mathbf{r}_i(t)$ are updated in the streaming steps, and their velocities $\mathbf{U}_i(t)$ are obtained through multi-particle collisions in the collision steps: $\mathbf{r}_i(t+\Delta t) = \mathbf{r}_i(t) + \Delta t \mathbf{U}_i(t)$, and $\mathbf{U}_i(t+\Delta t) = \mathbf{U}_i(t) + \mathbf{R} * \mathbf{U}_i(t)$. The stochastic rotational dynamics (SRD) is one of the most widely used MPCD algorithms in which the collision step consists of a random rotation \mathbf{R} of the relative velocities of the particles, i.e., $\Delta \mathbf{U}_i(t) = \mathbf{U}_i - \mathbf{U}$, in a collision cell, where \mathbf{U} is the mean velocity of all particles in a cell. Gompper et al. provided a review of several widely used MPC algorithms and recent applications of MPCD algorithm to study colloid and polymer dynamics as well as the behavior of vesicles and cells in hydrodynamic flow environments [36].

3.4.3 Dissipative particle dynamics (DPD): To reach larger time- and lengthscales, the dissipative particle dynamics (DPD) method uses a much coarser mapping, in which one site may represent many molecules in a small fluid volume [37-39]. There are three types of forces present in DPD models: a conserved soft repulsion force, pairwise dissipation forces, and pairwise random forces. The balance of dissipation and random forces provides the thermostat for the DPD model, and since this thermostat preserves the momentum of individual particles, these models provide correct hydrodynamic behavior. In addition to using a coarser mapping, DPD simulations use a longer timestep due to the use of soft repulsion forces. It is necessary to match the observed compressibility in a DPD simulation to the target fluid in order to study the phase behavior and interfacial tension of the model fluid. The DPD method has been applied to biological lipid bilayers, membrane fusion processes, and bilayers with proteins, and its connections to the mesoscale have been reviewed extensively [40, 41].

3.5 Continuum models based for fluid flows

We summarize direct numerical simulations (DNS) and lattice Boltzmann methods for solving transport equations.

3.5.1 Direct numerical simulation using finite element method (FEM): A finite element based arbitrary Lagrangian-Eulerian (ALE) technique can be used to directly solve equations such as (Eq. 3) handle the movement of single or multiphase domains including particle motions and fluid flow. An adaptive finite element mesh, generated by the Delaunay-Voronoi method, enables a significantly higher number of mesh points in the regions of interest (i.e., close to the particle and wall surfaces compared to the regions farther away). This feature also keeps the overall mesh-size computationally reasonable [42].

3.5.2 The lattice Boltzmann method (LBM): Vast number of applications of the lattice Boltzmann method (LBM) in simulating heat and mass transfer in fluids, particularly in complex geometries and with multi-components, have been demonstrated by previous researchers [43, 44]. This approach's primary strategy is to incorporate the microscopic physical interactions of the fluid particles in the numerical simulation and reveal the mesoscale mechanism of hydrodynamics. The LBM uses the density distribution functions $f(x_i, v_i, t)$ (similar to the Boltzmann or Liouville equations) to represent a collection of particles with the microscopic velocities v_i and positions x_i at time t , and model the propagation and collision of particle distribution taking the Boltzmann equations for flow and temperature fields into consideration. The LBM solves the discretized Boltzmann equation in velocity space through the propagation of the particle distribution functions $f(x, t)$ along with the discrete lattice velocities e_i and the collision operation of the local distributions to be relaxed to the equilibrium distribution f_i^0 . The collision term is usually simplified to the single-relaxation-time Bhatnagar-Gross-Krook (BGK) collision operator, while the more generalized multi-relaxation-time collision operator can also be adopted to gain numerical stability. The evolution equation for a set of particle distribution function with a single relaxation time is defined as:

$$f_i(x - \Delta x, t + \Delta t) = f_i(x, t) - (\Delta t / \tau) [f_i(x, t) - f_i^0(x, t)] + F_s, \quad (\text{Eq. 13})$$

where Δt is the time step, $\Delta x = \Delta t e_i$ is the unit lattice distance, and τ is the single relaxation time scale associated with the rate of relaxation to the local equilibrium, and F_s is a forcing source term introduced to account for the discrete external force effect. The macroscopic variables such as density and velocity, are then obtained by taking moments of the distribution function, i.e., $\rho = \sum_i f_i^{eq}$ and $\rho v = \sum_i f_i^{eq} e_i$. As explained earlier, through averaging the mass and momentum variables in the discrete Boltzmann equation, the continuity, and Navier-Stokes equations may be recovered.

3.5.3 Fluctuating hydrodynamics method: As noted in (Eq. 4), thermal fluctuations are included in the equations of hydrodynamics by adding stochastic components to the stress tensor as white noise in space and time as prescribed by the FHD method [12, 45]. Even though the original equations of fluctuating hydrodynamics are written in terms of stochastic partial differential equations, at a very fundamental level, the inclusion of thermal fluctuations always requires the notion of a mesoscopic cell in order to define the fluctuating quantities. The fluctuating hydrodynamic equations discretized in terms of finite element shape functions based on the Delaunay triangulation satisfy the fluctuation-dissipation theorem. The numerical

schemes for implementing the thermal fluctuations in the FHD equations are delicate to implement, and obtaining accurate numerical results is a challenging endeavor [46].

4. Parallel and high-performance computing

As noted earlier, a summary of the current capabilities of some of the multiphysics methods on current computing platforms is provided in Figure 1. In order to truly leverage the power of these methods in real-world applications, one needs to utilize parallel and HPC resources, which we discuss below. In current terms, high-performance computing (HPC) broadly involves the use of new architectures (such as GPU computing), computing in distributed systems, cloud-based computing, and computing in parallel to massively parallel platforms or extreme hardware architectures for running computational models (Figure 2, left). The term applies especially to systems that function with large floating-point operations per second (teraflops 10^{12} , petaflops 10^{15} , exaflops 10^{18}) regimes or systems requiring extensive memory. HPC has remained a sustained and powerful driving force for multiphysics modeling and scientific computing and central to applications in science, engineering, and medicine.

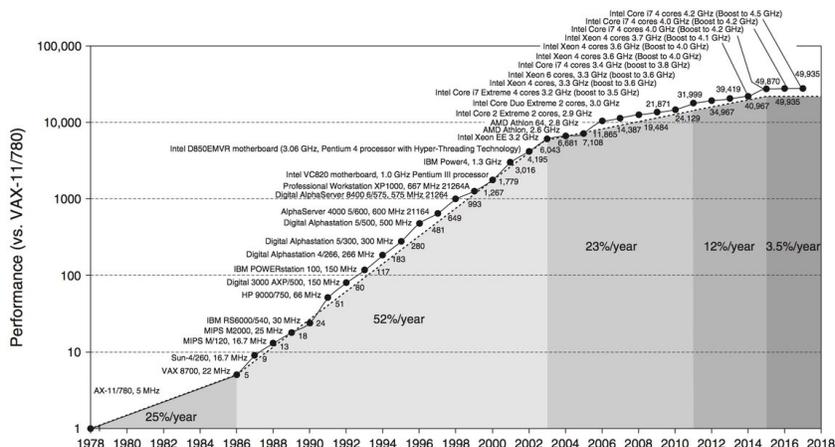
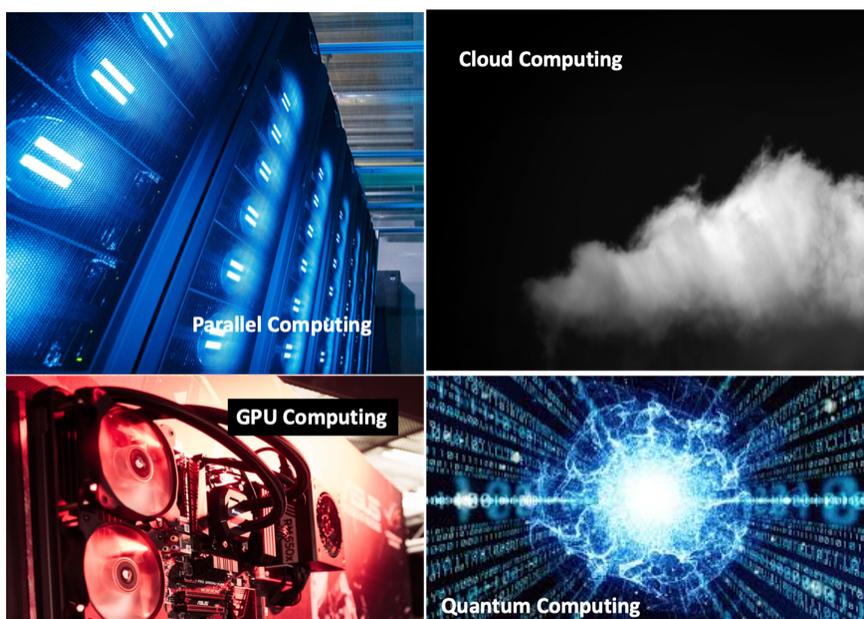


Figure 2 : HPC paradigms – current and future; Moore’s law and the slow-down due to the power wall.

A summary of historical developments in parallel and high-performance computing architectures is sketched in Table 3. A sketch of the early instruction for computation, including the concept of parallelism in computing, can be traced to Charles Babbage (see Table 3). Scientific computing has benefitted from the advances in chip architecture that led to the linear Moore’s law behavior for four decades from the 1980s-2010 (Figure 2, right). However, even during this golden age of increasing clock speeds and doubling of computational speed every 18 months in single-core architectures, high-performance computing broke the shackles of serial (and vectorized) computing to embrace parallel computing as a mainstream route to solve computational problems. The switch to parallel microprocessors is a game-changer in the history of computing [47] (see, <http://www.eecs.berkeley.edu/Pubs/TechRpts/2006/EECS-2006-183.html>). The advances in parallel hardware and software have torpedoed the advances in multiphysics and multiresolution simulations. This convergence of high-performance computing and multiscale modeling has transformed parallel algorithms (see Table 2), which are the engines of multiphysics modeling.

1842	Parallelism in computing; Menabrea, L. F. (1842). Sketch of the Analytic Engine Invented by Charles Babbage
1958-1970	Parallel computers: IBM Burroughs, Corp, Honeywell, Wilson, Gregory V. (1994). The History of the Development of Parallel Computing
1992	Message passing interface as a standard for communication across compute nodes in inherently and massively parallel architectures
2005	Establishment of multicore architecture by intel and others to circumvent the power wall inhibiting Moore’s law
2007	Release of CUDA: parallel computing platform and application programming interface (API) for graphics processing units (GPUs)

Table 3 : Historical milestones in parallel computing architectures

The paradigm shift in bringing the Babbage vision of parallelism to the common folk occurred with the change in emphasis from single instruction multiple data (SIMD) or embarrassingly parallel tasks where the same code is run multiple times in multiple processors to explore different parameters, to multiple instructions multiple data (MIMD) or inherently parallel tasks where the multiple processors work collectively to divide a given task by communicating with each other continually. The Message Passing Interface (MPI) (see Table 3) established a portable message-passing standard designed by a group of researchers from academia and industry to function on a wide variety of parallel computing architectures. The standard defines the syntax and semantics of a core of library routines useful to a wide range of users writing portable message-passing programs in C, C++, and Fortran. There are several well-tested and efficient implementations of MPI, many of which are open-source or in the public domain. These fostered the development of parallel software industry and encouraged the development of portable and scalable large-scale parallel applications utilizing the MIMD paradigm. Specifically, specific examples in the molecular dynamics of large systems or quantum chemistry codes could only be realized by shared memory and message passing architectures in the MIMD model.

4.1 Multicore architecture and the decline of Moore’s law

The linear trends in Figure 2, right ceases to hold beyond 2007 prediction due to the power wall in chip architecture. The industry was forced to find a new paradigm to sustain performance enhancement. The viable option was to replace the single power-inefficient processor with many efficient processors on the same chip, with increasing numbers of processors, or cores, each technology generation every two years. This style of the chip was labeled a multicore microprocessor. Hence, the leap to multicore is not based on a breakthrough in programming or architecture and is a retreat from building power-efficient, high-clock-rate, single-core chips [47]. The emergence of the multicore architecture in 2005, prompted shared memory architectures and the establishment of the application programming interface (API), OpenMP (Open Multi-Processing) standard, which supports multi-platform shared memory multiprocessing programming in C, C++, and Fortran (OpenMP.org).

One of the main drawbacks of MIMD platforms is the high cost of infrastructure. The alternative to MIMD platforms is single instruction multiple data (SIMD) architectures. With the increase of the computational

power and multicore options at the desktop level, and the low costs of the new processing architectures such as the graphical processing units, the attention of the scientific community has moved back to SIMD platforms [48]. The use of GPUs in scientific computing has exploded enabled by programming and instructional languages like CUDA (Compute Unified Device Architecture), a parallel computing platform and application programming interface (API) model created by Nvidia (<https://en.wikipedia.org/wiki/CUDA>). CUDA allows software developers and software engineers to use a CUDA-enabled graphics processing unit (GPU) for general-purpose processing – an approach termed GPGPU (General-Purpose computing on Graphics Processing Units). The CUDA platform is a software layer that gives direct access to the GPU’s virtual instruction set and parallel computational elements for compute kernels. Challenges for researchers utilizing HPC platforms and infrastructure range from understanding emerging new platforms to optimizing algorithms in massively parallel architectures to efficiently access and handle data at a large scale. The HPC technology is rapidly evolving and is synergistic yet complementary to the development of scientific computing: some useful links on HPC reviews, training, community, and resources are summarized in Table 4.

Parallel computing: The Landscape of Parallel Computing Research: A View from Berkeley Krste Asanović et al.: (https://www.pearc.org/)
GPU computing: GPU computing for systems biology, Lorenzo Dematté Davide Prandi, Brief Bioinform (2010) 11 (3): 3–10
Cloud computing: A scoping review of cloud computing in healthcare, Lena Griebel, Hans-Ulrich Prokosch, Felix Köpcke, BMC Health Services Research (2018) 18:1–12
HPC virtual workshops and virtual training: (https://cvw.cac.cornell.edu/topics)
HPC resources: Exascale computing project (https://www.exascaleproject.org/); The Extreme Science and Engineering Initiative (https://www.esandengineering.org/)
HPC conferences: PEARC (Practice & Experience in Adv. Res. Computing) (https://www.pearc.org/); Supercomputing Conference (https://www.supercomputing.org/)
HPC and industry: Intel – (https://www.intel.com/content/www/us/en/high-performance-computing/overview.html); IBM (https://www.ibm.com/press/us/en/industrycollaboration/041216.shtml)
Quantum computing: Andrew Steane, Rep. Prog. Phys., 61, 2; DOI: (10.1088/0034-4885/61/2/002) Quantum Computing: A Review of Progress and Prospects

Table 4 : Resources on HPC training, resources, community

In Table 2, we mentioned the development of parallel algorithms, which led to a transformation in multiphysics simulations: some examples include parallel matrix operations and linear algebra (<https://cvw.cac.cornell.edu/APC/>); parallel implementation of the N-body problem with short-range interactions (<https://cvw.cac.cornell.edu/APC/>); long-range interactions and the parallel particle mesh Ewald sum [49]; parallel Monte Carlo [50]; linear-scaling methods such as multipole expansion [49]; linear-scaling density functional theory [51]; parallel graph algorithms (<https://cvw.cac.cornell.edu/APC/>). As a specific example, we note that the N-body problem is an essential ingredient in MD. A common goal in MD of large systems is to perform sufficient sampling of the combinatorially large number of conformations available to even the simplest of biomolecules [52, 53]. In this respect, a potential disadvantage of molecular dynamics calculations is that there is an inherent limitation upon the maximum time step used for the simulation ([?] 2 fs). Solvated systems of biomolecules typically consist of 10^5 - 10^6 atoms. For such system sizes, with current hardware and software, simulation times extending into the tens of microseconds regime is an exceedingly labor-intensive and challenging endeavor that requires a combination of algorithmic enhancements as well as the utilization of high-performance computing hardware infrastructure. For example, cutoff distances reduce the number of interactions to be computed without loss of accuracy for short-range interactions but not for long-range (electrostatic) interactions; long-range corrections such as the particle mesh Ewald algorithm [54] along with periodic boundary conditions are typically implemented for maintaining accuracy. Parallelization techniques enable the execution of the simulations on supercomputing resources such as 4096 processors of a networked Linux cluster. Although a cluster of this size is a big investment, its accessibility is feasible through the extreme science and engineering discovery environment (XSEDE) for academic researchers. XSEDE resources (www.xsede.org) currently include petaflop of computing capability, and other US national laboratories such as the Oakridge are moving towards exascale computing (<https://www.exascaleproject.org>) [55]. Another approach, capitalizing on advances in hardware architecture, is creating custom hardware for MD simulations, and offers one-two orders of magnitude enhancement in performance; examples include MDGRAPE-3 [56, 57] and ANTON [58, 59]. Graphical processing unit (GPU) accelerated computation has

recently come into the forefront to enable massive speed enhancements for easily parallelizable tasks with early data indicating that GPU accelerated computing may allow for the power of a supercomputing cluster in a desktop, see e.g., [60, 61].

5. Multiscale modeling 1.0

The acceptance of multiphysics simulation techniques has helped bridge the gap between theory and experiment [62]. Electronic structure (quantum level or *ab initio*) simulations can reveal how specific molecules assume stable geometrical configurations and charge distributions when subject to a specific chemical environment. By examining the charge distributions and structure, it is possible to quantify and predict structural properties as well as chemical reactivity pertaining to the molecule, which is particularly pertinent when investigating novel materials. Although the quantum simulations provide a wealth of information regarding structure and reactivity, it is currently not possible to model much more than a few hundred atoms at most. Molecular dynamics simulations based on classical (empirical) force-fields can model hundreds of thousands of atoms for tens of microseconds in time. Since MD simulations can be set up at atomic resolution, they are uniquely suited to examine thermodynamic and statistical properties of (bio)materials: such properties include (but not limited to) Young’s modulus, surface hydration energies, and protein adsorption to different surfaces [63]. Coarse-grained or mesoscale simulations are used to bridge the gap between the atomistic scale of MD simulations and continuum approaches such as elasticity theory or hydrodynamics at the macroscale (i.e., milliseconds, millimeters and beyond) [62].

The ultimate purpose of multiscale modeling is to predict the macroscopic behavior from the first principles. Finding appropriate protocols for multiscale simulations is also challenging as either multiphysics simulations need to operate at multiple resolutions, or two or more multiphysics simulations need to be combined. In general, these are achieved via adaptive resolution schemes, coarse-graining, sequential multiscale modeling, concurrent multiscale modeling, and enhanced sampling schemes [18, 64], see Table 5.

Enhanced sampling methods [6]	Umbrella sampling [30]	Parallel tempering [6]	Metadynamics [65]	Path sampling [66]
Coarse graining methods [71, 72]	Structure matching method [73]	Force matching methods [74]	Energy matching meth	
Sequential multiscale methods	Parameter passing methods [78]	Particle to field passing [79]	Loosely coupled process flo	

Table 5 : recipes for multiscale modeling

The sequential approach links a series of computational schemes in which the operative methods at a larger scale utilize the coarse-grained (CG) representations based on detailed information attained from smaller-scale methods. Sequential approaches are also known as implicit or serial methods. The second group of multiscale approaches, the concurrent methods, are designed to bridge multiple individual scales in a combined model. Such a model accounts for the different scales involved in a physical problem concurrently and incorporates some sort of a handshaking procedure to communicate between the scales. Concurrent methods are also called parallel or explicit approaches. Another concept for multiscale simulations is adaptive resolution simulations. Finally, a number of advanced techniques allow for extending the reach of a single-scale technique such as MD within certain conditions. Such methods offer a route to temporal multiscale modeling through enhanced conformational sampling strategies. While these are a lot to review in detail, we summarize these methods into the subclasses followed by references (Table 5) and choose to highlight just some of the more foundational methods below. There is an entire journal dedicated to MSM, Multiscale Modeling and Simulation (<https://www.siam.org/journals/mms.php>).

5.1 Enhanced sampling methods

The second law of thermodynamics states that natural systems seek a state of minimum free energy at equilibrium. Thus, the computation of a system’s free energy is essential in comparing the results of simulation and experiment. Several different methods have been implemented for calculation of the free energy of various chemical and biomolecular systems, and here we will discuss three of the more commonly em-

methods such as stochastic path approach [95], nudged elastic band [96-98], finite temperature string [99], and transition path sampling [66, 100, 101], which each exploit the separation in timescales in activated processes, namely, the existence of a shorter time scale of relaxation at the kinetic bottle neck or the transition state (τ_{relax}), in comparison to a much longer timescale of activation at the transition state itself (τ_{TS}). Below, we review the path-based method of transition path sampling.

Transition path sampling (TPS) [66, 100] aims to capture rare events (excursions or jumps between metastable basins in the free energy landscape) in molecular processes by essentially performing Monte Carlo sampling of dynamics trajectories; the acceptance or rejection criteria are determined by selected statistical objectives that characterize the ensemble of trajectories. In transition path sampling, time-reversible MD trajectories in each transition state region are harvested using the shooting algorithm [101] to connect two metastable states via a Monte Carlo protocol in trajectory space. Essentially, for a given dynamics trajectory, the state of the system (i.e., basin A or B) is characterized by defining a set of order parameters $\chi = [\chi_1, \chi_2, \dots]$. Each trajectory is expressed as a time series of length τ . To formally identify a basin, the population operator $h_A = 1$ if and only if a particular molecular configuration associated with a time t of a trajectory belongs to basin A; otherwise $h_A = 0$. The trajectory operator $H_B = 1$ if and only if the trajectory visits basin B in duration τ , i.e., there is at least one time-slice for which $h_B = 1$; otherwise $H_B = 0$. The idea in TPS is to generate many trajectories that connect A to B from one such existing pathway. This is accomplished by a Metropolis algorithm that generates an ensemble of trajectories $[\chi]$ according to a path action $S[\chi]$ given by: $S[\chi] = \rho(0) h_A(\chi_0) H_B[\chi]$, where $\rho(0)$ is the probability of observing the configuration at $t=0$ ($\rho(0) = \exp(-E(0)/k_B T)$, in the canonical ensemble). Trajectories are harvested using the shooting algorithm [101]: a new trajectory $\chi^{*\tau}$ is generated from an existing one χ^τ by perturbing the momenta of atoms at a randomly chosen time t in a symmetric manner [101], i.e., by conserving detailed balance. The perturbation scheme is symmetric, i.e., the probability of generating a new set of momenta from the old set is the same as the reverse probability. Moreover, the scheme conserves the equilibrium distribution of momenta and the total linear momentum (and, if desired, the total angular momentum). The acceptance probability implied by the above procedure is given by $P_{\text{acc}} = \min(1, S[\chi^*]/S[\chi])$. With sufficient sampling in trajectory space, the protocol converges to yield physically-meaningful trajectories passing through the true transition state (saddle) region.

5.2 Coarse graining

Coarse-grained molecular dynamics simulations employ intermediate resolution in order to balance chemical detail with system size. They offer sufficient size to study membrane-remodeling events while retaining the ability to self-assemble. Because they are capable of simulating mesoscopic length scales, they make contact with a wider variety of experiments. A complete coarse-grained model must include two components: a mapping from atomistic structures to coarse-grained beads and a set of potentials that describe the interactions between beads. The former defines the geometry or length scale of the resulting model, while the latter defines the potential energy function or the force field. The parameterization of the force field is essential to the performance of the model, which is only relevant insofar as it can reproduce experimental observables. Here we will describe the characteristic methods for developing CGMD models, namely the bottom-up structure- and force-matching and top-down free energy-based approaches. We note that excellent reviews have been written on coarse-grained methods with applications in other fields such as polymer physics, see, e.g., [72].

5.2.1 Structure and energy matching in the CMM-CG model:

Klein and co-workers developed a coarse-grained model for phospholipid bilayers by matching the structural and thermodynamic properties of water, hydrocarbons and lipid amphiphile to experimental measurements and all-atom simulations [102]. The resulting force field, titled CMM-CG, has been used to investigate a range of polymer systems as well as those containing nonionic liquids and lipids. Classic coarse-grained methods propose pair potentials between CG beads according to the Boltzmann inversion method. In this method, a pair correlation function, or radial distribution function (RDF) $g(r)$ defines the probability of finding a particle at distance r from a reference particle such that the conditional probability of finding the particle is $\rho(r) = \rho g(r)$, where ρ is the average number density of the fluid. The potential of mean force (PMF) between CG beads is then estimated by (Eq.

18) where $g_{aa}(r)$ is the RDF measured from atomistic simulation, and α_n is a scaling factor (corresponding to the n th iteration of the estimate) designed to include the effect of interactions with the (necessarily) heterogeneous environment.

$$V_n(r) = \alpha_n \left\{ -k_B T \ln(g_{aa}(r)) \right\}$$

(Eq. 18)

The Boltzmann inversion method is iteratively corrected according to (Eq. 19) to correct the tabulated potentials until the pair-correlation functions for the atomistic and coarse-grained systems agree.

$$V_{n+1}(r) = V_n(r) + k_B T \ln \frac{g_n(r)}{g_{aa}(r)}$$

(Eq. 19)

5.2.2 Force matching with the MS-CG model : The method of force-matching provides a rigorous route to developing a coarse-grained force field directly from forces measured in all-atom simulations. In so far as the multi-body coarse-grained PMF is derived from structure factors that depend on temperature, pressure, and composition, they cannot be transferred to new systems. To avoid this problem, the force-matching approach proposes a variational method in which a coarse-grained force field is systematically developed from all-atom simulations under the correct thermodynamic ensemble [103]. In the statistical framework developed by Izvekov and Voth [103-105], it is possible to develop the exact many-body coarse-grained PMF from a trajectory of atomistic forces with a sufficiently detailed set of basis functions.

The method starts with a collection of sampled configurations from an atomistic simulation of the target system and calculate the reference forces between atoms of a particular type. After decomposing their target force into a short-ranged part approximated by a cubic spline and a long-ranged Coulomb part one solves the overdetermined set of linear equations given by (Eq. 20):

$$\sum_{\beta=1}^K \sum_{j=1}^{N_{\beta}} \left(-f(r_{\alpha i l, \beta j l}, \{r_{\alpha \beta, \kappa}\}, \{f_{\alpha \beta, \kappa}\}, \{f''_{\alpha \beta, \kappa}\}) - \frac{q_{\alpha \beta}}{r_{\alpha i l, \beta j l}^2} \right) \mathbf{n}_{\alpha i l, \beta j l} = \mathbf{F}_{\alpha i l}^{\text{ref}}$$

(Eq. 20)

In equation (Eq. 20), the $r_{\alpha \beta, \kappa}$ correspond to the spline mesh at points κ for pairs of atoms of type α and β , while f and f'' are spline parameters that ensure continuous derivatives $f'(r)$ at the mesh points and define the short-ranged part of the force. The subscript α_i labels the i th atom of type α in the l th sampled atomic configuration. Solving these equations minimizes the Euclidean norm of vectors of residuals, and can be solved on a minimal set of atomistic snapshots using a singular value decomposition (SVD) algorithm. By adding the Coulomb term to the short-ranged potential above, this technique allows for the inclusion of explicit electrostatics. The MS-CG model reproduces site-to-site RDFs from atomistic MD simulations in the as well as the density profile perpendicular to the bilayer normal in DMPC bilayers.

5.2.3 The energy-based approach of the Martini force field: The Martini force field developed by Marrink and co-workers eschews systematic structure-matching in pursuit of a maximally transferable force field which is

parameterized in a “top-down” manner, designed to encode information about the free energy of the chemical components, thereby increasing the range of thermodynamic ensembles over which the model is valid [75].

The Martini model employs a four-to-one mapping of water and non-hydrogen atoms onto a single a bead, except in ring-like structures, which preserve geometry with a finer-scale mapping. Molecules are built from relatively few bead types, which are categorized by polarity (polar, nonpolar, apolar, and charged). Each type is further distinguished by hydrogen bonding capabilities (donor, acceptor, both, or none) as well as a score describing the level of polarity. Like the CMM-CG and MS-CG models, Lennard-Jones parameters for nonbonded interactions are tuned for each pair of particles. These potentials are shifted to mimic a distance-dependent screening effect and increase computational efficiency. Charged groups interact via a Coulomb potential with a low relative dielectric for explicit screening. This choice allows the use of full charges while reproducing salt structure factors seen in previous atomistic as well as the hydration shell identified by neutron diffraction studies. Nonbonded interactions for all bead types are tuned to semi-quantitatively match measurements of density and compressibility. Bonded interactions are specified by potential energy functions that model bonds, angles, dihedrals, and impropers with harmonic functions, with relatively weak force constants to match the flexibility of target molecules at the fine-grained resolutions. The Martini force field’s defining feature is the selection of nonbonded parameters that are optimized to reproduce thermodynamic measurements in the condensed phase. Specifically, the Martini model semi-quantitatively reproduces the free energy of hydration, the free energy of vaporization, and the partitioning free energies between water and a collection of organic phases, obtained from the equilibrium densities in both phases [75].

5.2.4 Structure-based coarse-grained protein modeling : While coarse-grained simulations have difficulty reproducing secondary structural transformations, it is possible to recover accurate conformational sampling by a reverse-transformation from the CGMD level to the atomistic one. atomistic simulations of back-mapped CGMD structures can recover the conformational properties of the original atomistic system. In this procedure, back-mapped atoms are randomly placed near their corresponding coarse-grained bead. The center of mass of these atoms is then restrained to the position of the coarse-grained bead. The system may be relaxed by a simulated annealing procedure to minimize large or unphysical forces, stochastically sample the conformation space, and gradually introduce inter- and intra-molecular potentials consistent with the all-atom model. This method has been used to generate atomistic structures of simple peptides and transmembrane proteins from coarse-grained trajectories. The back-mapping procedure also quantifies the information loss from coarse-graining, providing a useful way to validate a CG model against a more robust atomistic force field or extend a CG trajectory to include greater detail. Elastic network models have found numerous applications in flexible fitting methods, which add detail to low-resolution cryo-EM measurements [106].

5.3 Minimal coupling methods

Minimal coupling methods minimize explicit and concurrent communication across scales via a variety of clever algorithmic or software architecture tricks and represent a power repertoire of multiscale methods. There are numerous techniques in this popular category, and we chose not to delve into any of them in detail. A few of the methods in these categories are listed along with their references in Table 5 under Sequential multiscale methods and Adaptive resolution methods. Sequential methods involve computing a property or a constitutive relationship at one (typically the molecular) scale and employing (either pre- or on-the-fly-) computed values in the other (typically the continuum scale) [65,70,71].

5.4 Concurrent multiscale methods

The concurrent approaches couple two or more methods and execute them simultaneously with continuous information transfer across scales in contrast to the minimal coupling methods which attempt to do the opposite. In this class of methods, the behavior at each scale depends strongly on the phenomena at other scales. A successful algorithm in the concurrent method implements a smooth coupling between the scales. In concurrent simulations, often, two distinct domains with different scales are linked together through a

buffer or overlap region called the handshake region [18].

5.4.1 Quantum mechanics molecular mechanics (QM/MM) Simulations: An example of concurrent include mixed quantum mechanics/ molecular mechanics (QM/MM) methods combining MD using the empirical force-field approach with electronic structure methods [19, 20, 107] to produce a concurrent multiscale method [76, 108-119]. In the QM/MM simulations, the system is sub-divided into two sub-regions, the quantum mechanical sub-region (QM region) where the reactive events take place, and the molecular mechanical sub-region (which provides the complete environment around the reactive chemistry) [109, 111]. Since electronic structure methods are limited by the number of atoms they can handle (typically 50-500), the QM sub-region is restricted to a small number of atoms of the total system. For example, in an enzymatic system, the quantum region can consist of Mg^{2+} ions, water molecules within 3 Å of the Mg^{2+} ions, parts of the substrate molecules, and the catalytic amino acid residues (such as aspartic acids). The remaining protein and solvent molecules are treated classically using the regular classical force-field.

In QM/MM simulations, wave function optimizations are typically performed in the quantum (or QM) sub-region of the system using an electronic structure method such as density functional theory (DFT) [20]. In this step, the electrostatic coupling between the QM and the MM sub-regions is accounted for: i.e., the charges in the MM sub-region are allowed to polarize the electronic wave functions in the QM sub-region. The forces in the quantum sub-region are calculated using DFT on-the-fly, assuming that the system moves on the Born-Oppenheimer surface [111, 120]. That is, we assume a clear timescale of separation between the electronic and nuclear degrees of freedom and the electronic degrees of freedom are in their ground state around the instantaneous configurations of the nuclei. The forces on the classical region are calculated using a classical force-field. Besides, a mixed Hamiltonian (energy function) accounts for the interaction of the classical and the quantum sub-regions. For example, since the QM/MM boundary often cuts across covalent bonds, one can use a link atom procedure [114] to satisfy the valences of broken bonds in the QM sub-region. Also, bonded terms and electrostatic terms between the atoms of the QM region and those of the classical region are typically included [112].

From a practitioner’s stand-point, QM/MM methods are implemented based on existing interfaces between the electronic structure and the molecular dynamics programs; one example implementation is between GAMESS-UK [121] (an *ab-initio* electronic structure prediction package) and CHARMM [25]. The model system can then be subjected to the usual energy minimization and constant temperature equilibration runs at the desired temperature using the regular integration procedures in operation for pure MM systems; it is customary to carry out QM/MM dynamics runs (typically limited to 10-100 ps because of the computationally intensive electronic structure calculations) using a standard 1 fs time step of integration. The main advantage of the QM/MM simulations is that one can follow reactive events and dissect reaction mechanisms in the active site while considering the explicit coupling to the extended region. In practice, sufficient experience and care is needed in the choices of the QM sub-region, and the many alternative choices of system sizes, as well as the link-atom schemes, need to be compared to ensure convergence and accuracy of results [112]. The shorter length of the dynamics runs in the QM/MM simulations (ps) relative to the MM MD simulations (ns) implies that sufficiently high-resolution structures are usually necessary for setting up such runs as the simulations only explore a limited conformational space available to the system. Another challenge is an accurate and reliable representation of the mixed QM/MM interaction terms [115]. These challenges are currently being overcome by the suitable design of next-generation methods for electronic structure and molecular mechanics simulations [51, 122]. Other examples of concurrent methods linking electronic structure and or molecular mechanics scales include Car Parrinello molecular dynamics (CPMD) [123, 124] and mixed molecular mechanics/ coarse-grained (MM/CG) [77, 125].

5.4.2 Linking atomistic and continuum models : In several applications involving solving continuum equations in fluid and solid mechanics, there is a need to treat a small domain at finer (often molecular or particle-based resolution) to avoid sharp fronts or even singularities. In such cases linking atomistic and continuum domains using bridging algorithms are necessary. A class of algorithms that realize this challenging integration have been reviewed in [18]: examples include the quasicontinuum approach, finite-element/ atomistic

method, bridging scale method, and the Schwartz inequality method [126, 127], which all employ domain decomposition bridging by performing molecular scale modeling in one (typically a small domain) and integrating it with continuum modeling in an adjoining (larger) domain, such that certain constraints (boundary conditions) are satisfied self-consistently at the boundary separating the two domains. Such approaches are useful for treating various problems involving contact lines.

5.5 Field-based coarse-graining

For specific systems such as nanoparticle and nanofluid transport, both molecular interactions (due to biomolecular recognition) and hydrodynamic interactions (due to fluid flow and boundary effects) are significant. The integration of disparate length and time scales does not fit traditional multiscale methods. The complexity lies in integrating fluid-flow and memory for multiphase flow in complex and arbitrary geometries, while simultaneously including thermal and stochastic effects to simulate quasi-equilibrium distributions correctly to enable receptor-ligand binding at the physiological temperature. This issue is ubiquitous in multivalent binding or adhesive interactions between nanoparticles and cells or between two cells. Bridging the multiple length scales (from meso to molecular) and the associated time scales relevant to the problem is essential to success herein. Multiple macroscopic and mesoscopic time scales governing the problem include (i) hydrodynamic time scale, (ii) viscous/Brownian relaxation time scale, and (iii) Brownian diffusion time scale.

5.5.1 Memory function approach to coarse-graining with hydrodynamic interactions: In the description of the dynamics of nanosized Brownian particles in an bounded and unbounded fluid domains the memory functions decay with algebraic correlations as enumerated by theoretical and computational studies [46, 85, 128]. The equation of stochastic motion for each component of the velocity of a nanoparticle immersed in a fluid in bounded and unbounded domains takes the form of a generalized Langevin equation (GLE) of the form of (Eq. 9); to account for hydrodynamic interaction, a composite GLE was introduced [129, 130].

$$\begin{aligned}
 M \frac{dU(x, t)}{dt} = & -6\pi\eta a\beta U(x, t) - A_1(t) \int_{-\infty}^t |t - t'|^{-3/2} U(x, t') dt' \\
 & - A_2(t) \int_{-\infty}^t |t - t'|^{-5/2} U(x, t') dt' - \frac{dF}{dx} + R(t).
 \end{aligned}$$

(Eq. 21)

Here M is the added mass, and β is the geometric factor with wall effect corrections, the integrands include the memory functions associated with the velocity autocorrelation functions in different domains (lubrication, bulk, and near-wall regimes in order of the first three terms on the right-hand side). The fourth term on the right-hand side is the force from other thermodynamic potentials, same as $F(S)$ in (Eq. 7), and the fifth term is the random force term with colored noise to be consistent with the fluctuation-dissipation theorem for composite GLE [129, 130].

Effect of molecular forces is introduced as forcing functions in the GLE [129] and the effect of multiple particles including multiparticle HI can be introduced via density functional theory-based treatments [82, 83] to define $F(S)$ from hydrodynamic and colloidal effects in addition to the specific contributions from molecular forces. If the memory functions are unknown, they can be obtained via deterministic approaches by solving the continuum hydrodynamic equations numerically [46, 128]. These disparate hydrodynamic fields and molecular forces can be integrated into a single GLE to realize a unified description of particle dynamics under the influence of molecular and hydrodynamic forces [85]. Another approach to integrating these forces is via the Fokker Planck approach using the sequential multiscale method paradigm [131].

6. Multiscale modeling 2.0

6.1 Integrating MSM and ML to elucidate the emergence of function in complex systems

We are riding the wave of a paradigm shift in the development of MSM methods due to rapid development and changes in HPC infrastructure (see Figure 2) and advances in ML methods. Thus, MSM and HPC have emerged as essential tools for modeling complex problems at the microscopic scales with a focus on leveraging the structured and embedded physical laws to gain a mechanism-based understanding. This success notwithstanding, the design of new MSM algorithms in coupling different scales, data utilization, and their implementation on HPC is becoming increasingly cumbersome in the face of heterogeneous data availability and rapidly evolving HPC architectures and platforms. On the other hand, while purely data-driven models of molecular and cellular systems spawned by the techniques of data science [132-134], and in particular, ML methods including deep learning methods [135-137], are easy to train and implement, the underlying model manifests as a black-box. This general approach taken by the ML community is well suited for classification, learning, and regression problems, but suffers from limitations in interpretability and explainability, especially when mechanism-based understanding is a primary goal. There lies a vast potential in combining MSM, HPC, and ML methods with their complementary strengths [4]. MSM models are routinely coupled together by appropriately propagating information across scales (see section 5), while the ever-increasing advances in hardware capabilities and high-performance software implementations allows us to study increasingly more complex phenomena at a higher fidelity and higher resolution. While much of the discussion thus far has been focused on MSM and HPC methods, the progress and potential in integrating MSM and ML are discussed below and represent the forefront of emerging MSM research, in which we discuss a few emerging integrative approaches to combine ML and MSM.

6.2 Integrators and autotuning

Over the past two decades, MSM has emerged into a promising tool to build in-silico predictive models by systematically integrating knowledge from the tissue, cellular, and molecular level. Depending on the scale of interest, governing equations in each scale of the MSM approaches may fall into two categories, ordinary differential equation-based, and partial differential equation-based approaches. Examples include molecular dynamics [21], coarse-grained mesoscale models [71], lattice Boltzmann methods [43], immersed boundary methods [138], as well as classical finite element approaches [139]. ML-based methods can speed up, optimize, and autotune several of the existing solvers for multiphysics simulations [140, 141].

6.3 ML-enabled MSM

As noted earlier, one of the main objectives of MSM is to couple the physics at different scales using bridging algorithms that pass information between two scales, such as in QM/MM, MM/CG, CG/CM, and field-based methods discussed in section 5. However, the implementation of this methodology on parallel supercomputing HPC architectures is complicated and cumbersome. To address this significant limitation in implementation, we advocate for an ML-enabled integration or bridging of scales as a viable approach to develop the next generation of MSM methods to achieve maximal efficiency and flexibility in integrating scales.

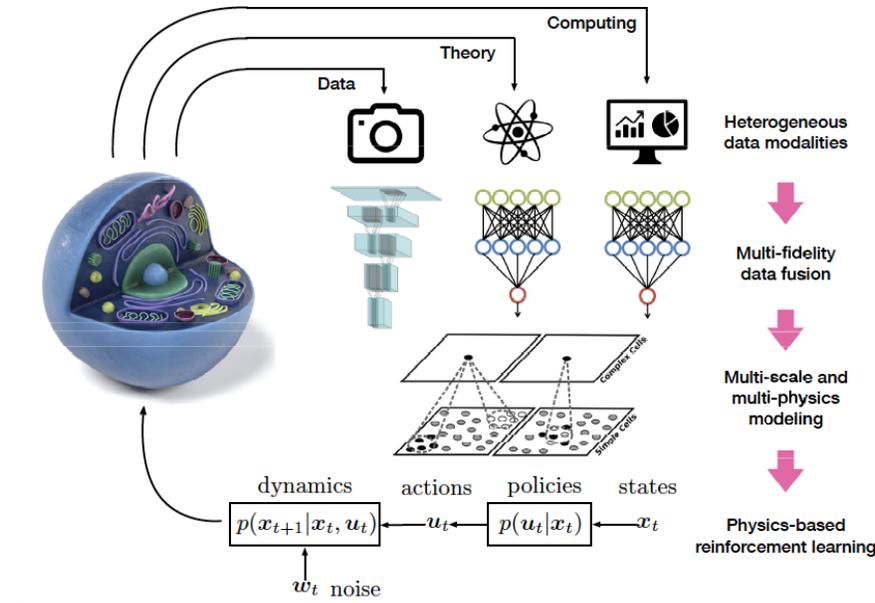


Figure 3 . The proposed synergy of multiscale and machine learning aspires to (i) accelerate the prediction of large-scale computational models, (ii) discover interpretable models from irregular and heterogeneous data of variable fidelity, and (iii) guide the judicious acquisition of new information towards elucidating the emergence of function in biological systems.

Here one can leverage ongoing developments in ML to accelerate the prediction of large-scale computational models. As a viable path forward, ML workflow can be implemented in three steps (see Figure 3) [4]: (1) Train deep neural network (NN) encoders that connect properties of the MSM model at scale 1 (e.g., MD or MC) to those at scale 2 (e.g., continuum model), using explicit MSM computations overextended parameter sets to cover all possible conditions; (2) implement the encoders in place of the coupling algorithms to bridge scales 1 & 2; (3) ensure that the properties profiles obtained from the two scales match by defining a cost-function that constrains the training of NNs in (1). The NN-based coupling of scales is expected to be robust, computationally efficient for MSM algorithms.

One challenge is to discover interpretable models from heterogeneous data of variable fidelity, and guide the judicious acquisition of new information towards elucidating the emergence of function in biological systems. This challenge can be addressed by subjecting the entire MSM model to contemporary data science and statistical methodologies, i.e., [142] sensitivity [1], evolvability [2], and robustness [143] analyses, uncertainty quantification [144], multi-fidelity modeling [145], and pattern discovery and model reduction [146].

6.4. Physics-informed neural networks

Can we use prior physics-based knowledge to avoid overfitting or non-physical predictions? From a conceptual point of view, can we supplement ML with a set of known physics-based equations, an approach that drives MSM models in engineering disciplines? While data-driven methods can provide solutions that are not constrained by preconceived notions or models, their predictions should not violate the fundamental laws of physics. There are well-known examples of deep learning neural networks that appear to be highly accurate but make highly inaccurate predictions when faced with data outside their training regime, and others that make highly inaccurate predictions based on seemingly minor changes to the target data [147]. To address this ubiquitous issue of purely ML-based approaches, numerous opportunities to combine machine learning and multiscale modeling towards a priori satisfying the fundamental laws of physics, and, at the same time, preventing overfitting of the data.

A potential solution is to combine deterministic and stochastic models. Coupling the deterministic governing equations MSM models — the balance of mass, momentum, and energy — with the stochastic equations of systems biology and biophysical systems — cell-signaling networks or reaction-diffusion equations — could help guide the design of computational models for otherwise ill-posed problems. Physics-informed neural networks (PINN) [148] is a promising approach that employs deep neural networks and leverages their well-known capability as universal function approximators [149]. In this setting, we can directly tackle nonlinear problems without the need for committing to any prior assumptions, linearization, or local time-stepping. PINNs exploit recent developments in automatic differentiation [150] to differentiate neural networks concerning their input coordinates and model parameters to obtain physics informed neural networks. Such neural networks are constrained to respect any symmetry, invariance, or conservation principles originating from the physical laws that govern the observed data, as modeled by general time-dependent and nonlinear partial differential equations. This construction allows us to tackle a wide range of problems in computational science and introduces a potentially disruptive technology leading to the development of new data-efficient and physics-informed learning machines, new classes of numerical solvers for partial differential equations, as well as new data-driven approaches for model inversion and systems identification.

6.5. Deep neural network algorithms inspired by statistical physics and information theory

Large amounts of data, cheap computation, and efficient algorithms are driving the impressive performance and adoption of robust deep learning architectures. However, building, maintaining, and expanding these systems is still decidedly an art and requires a lot of trial and error. Learning and inference methods have a history of being inspired by and derived from the principles of statistical physics and information theory [151, 152]. We summarize examples to advance this theme to derive NN algorithms based on a confluence of ideas in statistical physics and information theory [153] and to feed them back into core MSM methods by prescribing new computational techniques for deep neural networks. (A) Generalization in deep NN: the approach utilizes algebraic topology [154, 155] to characterize the space of reachable functions using stochastic dynamics on data in order to build computationally efficient architectures and algorithms to train them [156-158]. (B) Characterizing the quality of representations and the performance of encoders, decoders: Recent works have proposed to exploit principles of representation learning to formulate variational approaches for the assessment of performance in deep learning algorithms [16] that provide guarantees on the performance of the final model.

6.6 ML-enhanced conformational sampling

Advances in ML-derived force fields are promising to revolutionize classical simulations by directly defining energy landscapes from more accurate quantum mechanical simulations [159, 160]. Besides, in particle-based simulations of MSM, the efficient sampling of high-dimensional conformational spaces constitutes a significant challenge in the computational molecular sciences limiting the longtime molecular dynamics (MD) simulations of molecular systems in biophysical chemistry and materials science. Combining MD simulations with ML can provide a powerful approach to address the challenges mentioned above [161]. The last decade has seen significant advances in the use of electronic structure calculations to train ML potentials for atomistic simulations capable of reaching large systems sizes and longtime scales with accurate and reliable energies and forces. More recently, ML approaches have proved useful in learning high-dimensional free energy surfaces [162, 163], and in providing a low dimensional set of collective variables or CVs [164]. Some examples are discussed below.

6.6.1 Boltzmann generators: The primary difficulty in sampling physical realizations or microstates of the system from the Boltzmann distribution lies in the nature of the potential energy. In large, complex systems, the conformational space holds the positions of hundreds of thousands to millions of atoms. The potential energy should be viewed as a vast, rugged landscape in this high-dimensional space characterized by an exponentially large number of low-energy regions or minima, all separated by ridges. It is now possible to train a deep neural network to learn a transformation from the conformational space to another variable-space such that in this new space, the variables are distributed according to simple distributions such as

the Gaussian distribution. One can back-map to the original space through inverse transformation onto a high-probability region of the original conformational space [161].

6.6.2 ML-enabled conformational enhanced sampling : The choice of appropriate collective variables (CVs, aka order parameters in the earlier section) for enhanced sampling methods such as metadynamics is still a challenge. Recent advances have enabled ML tools of supervised learning to define appropriate CVs. The pipeline flow for such supervised machine learning methods utilizes ML-identified features from (A) as CVs for enhanced sampling simulations [165]. Another choice for ML-identified CV is through the use of variational autoencoders [166], which are deep NNs that perform dimensionality reduction similar to principal component analysis. The neural encoder takes a high dimensional input vector and outputs a lower-dimensional output vector. The neural decoder then takes the latent variable as input and attempts to reconstruct the original high dimensional input using standard optimization techniques of a loss function [167].

6.6.3 ML-enhanced adaptive path sampling: While enhanced sampling methods are efficient in exploring the landscape based on pre-defined CVs, one often discovers new variables during sampling. In such scenarios, new variables that become relevant are often orthogonal to the original CVs, and it is impossible to incorporate such variables adaptively into the free energy landscape. One approach is to combine enhanced sampling such as metadynamics with path sampling and utilize the newly identified CVs in the path sampling through a path action [168]. In this approach, path sampling is pursued by adaptively modifying the path action, and the free energy landscape based on the original CVs can be refined iteratively. The path action approach is also easily customized to include other ML strategies such as reinforcement learning to guide the system through non-Boltzmann paths.

6.7. Ab-initio methods using quantum computing

Quantum computers hold promise to enable efficient simulations of the properties of molecules and materials; however, at present, their abilities are limited due to a limited number of qubits that can be realized. In the near-term, the throughput of quantum computers is limited by the small number of qubits available, which prohibits large systems. It is more practical to develop hybrid quantum-classical methods where the quantum computation is restricted to a small portion of the system; for example, molecules where an active region requires a higher level of theoretical accuracy than its environment. Galli et al. outline a quantum embedding theory for the calculation of strongly-correlated electronic states of active regions, with the rest of the system described within density functional theory [169]. The authors demonstrate the efficacy of the method by investigating defect quantum bits in semiconductors that are of great interest for quantum information technologies. The calculations are performed on quantum computers and show that they yield results in agreement with those obtained with exact diagonalization on classical architectures, paving the way to simulations of realistic materials on near-term quantum computers.

7. Conclusion

Amidst the explosion of data in all walks of science, engineering, biology and biomedical science, it is useful to seek an interpretable basis for the emergence of function. How can geometry, physics, and engineering best inform biology or lead to the discoveries of new functional advanced materials? The complex multiscale interactions that characterize the dynamic behavior of biological systems [170] and advanced materials have limited our ability to understand the fundamental mechanisms behind the emergence of function to relatively idealized systems [171].

ML integrated with MSM is poised to enhance the capabilities of standard MSM approaches profoundly, particularly in the face of increasing problem complexity and data intensiveness. The contemporary research problems warrant an interdisciplinary environment to tackle emerging scientific and technological grand-challenge problems that carry substantial societal impact. Research projects, while posed across varied application domains in the broad STEM field, often have common features: (1) the problem/solution spans diverse length and timescales and benefits from MSM, (2) ML methods integrate into the MSM methods to define the new approaches at the frontiers of MSM development, (3) tools of data science are effectively

leveraged to integrate experimental data with the proposed model, and (4) the implementation of the model will utilize HPC methods and/ or platforms. Foundational training for future scholars should ideally provide: (1) working knowledge in fundamental science and modeling methodologies at multiple lengths and timescales spanning the molecular to process scales; (2) the requisite skills to integrate, and couple multiple scales into a multiscale paradigm; (3) learnings to exploit elements of data science, including machine learning methods and tools of data integration from cloud-based, data-rich repositories in order to validate and test computational models and software; (4) learnings to combine the rich tools of ML with MSM methods to define the next-generation of MSM methods; (5) experience to adopt, and implement best practices in software architecture to leverage modern computational infrastructure and develop efficient sustainable codes. With these foundations and skill-sets in the arsenal of the emerging researcher, the potential to blend MSM, HPC, and ML presents opportunities for unbound innovation and represents the future of MSM and explainable ML that will likely define the fields in the 21st century.

Acknowledgments

I acknowledge insightful discussions from students of BE559 multiscale modeling of chemical and biological systems at the University of Pennsylvania, where the topics in this review are discussed as a one-semester course. I would like to thank my colleagues at the Penn Institute for Computational Science for insightful discussions on multiscale modeling over the years. I would like to take this opportunity to acknowledge the profound influence Keith E. Gubbins has had, as my PhD thesis advisor, as a professional mentor throughout my faculty career and through his books, Reid and Gubbins – applied statistical mechanics, and Gray and Gubbins – theory of molecular fluids 1 and 2. Funding for this work was provided in part through various grants from NSF, NIH, EU, and XSEDE. This report reflects a very subjective and personal journey, and I regret and apologize for not citing all of the influencers in the discipline of multiscale modeling.

References

1. Sobol, I.M., *Sensitivity estimates for nonlinear mathematical models*. Mathematical modelling and computational experiments, 1993.1 (4): p. 407–414.
2. Brown, K.S. and J.P. Sethna, *Statistical mechanical approaches to models with too many unknown parameters*. Phys. Rev. E, 2003.68 : p. 021904.
3. Ghosh, A., *A Heterogeneous and Multiscale Modeling Framework to Develop Patient-Specific Pharmacodynamic Systems Models in Cancer*. PhD Thesis, University of Pennsylvania, 2019.
4. Alber, M., et al., *Integrating machine learning and multiscale modeling—perspectives, challenges, and opportunities in the biological, biomedical, and behavioral sciences*. npj Digital Medicine, 2019. 2 (1): p. 115.
5. van Kampen, N.G., *Stochastic processes in physics and chemistry* . 1992, Amsterdam: North-Holland.
6. Frenkel, D. and B. Smit, *Understanding Molecular Simulations. From Algorithms to Applications* . 1996, San Diego, CA: Academic Press.
7. Reed, T.M. and K.E. Gubbins, *Applied Statistical Mechanics: Thermodynamic and Transport Properties of Fluids* . 1991: Butterworth-Heinemann.
8. Cáceres, M.O. and A.K. Chattah, *On the Schrödinger-Langevin picture and the master equation*. Physica A: Statistical Mechanics and its Applications, 1996. 234 (1): p. 322-340.
9. Jackson, J.D., *Classical electrodynamics* . 1999: Third edition. New York : Wiley, [1999] ©1999.
10. Chapman, S. and T.G. Cowling, *The Mathematical Theory of Non-uniform Gases* . 1991: Cambridge Mathematical Library.
11. Zwanzig, R. and M. Bixon, *Hydrodynamic Theory of the Velocity Correlation Function*. Physical Review A, 1970. 2 (5): p. 2005-2012.

12. Landau, L.D. and E.M. Lifshits, *Fluid mechanics* . 1987: Pergamon Press.
13. Chandler, D., *Introduction to modern statistical mechanics* . 1987, New York, NY: Oxford University Press.
14. Chaikin, P.M. and T.C. Lubensky, *Principles of condensed matter physics* . 1995, Cambridge ; New York: Cambridge University Press. xx, 699 p.
15. Kubo, R., *The fluctuation-dissipation theorem*. Reports on Progress in Physics, 1966. **29** (1): p. 255-284.
16. Crooks, G.E., *Excursions in statistical dynamics*. PhD Thesis, University of California, Berkeley, 1999.
17. Balakrishnan, V., *Elements of Nonequilibrium Statistical Mechanics* . 2008: CRC Press.
18. Gooneie, A., S. Schuschnigg, and C. Holzer, *A Review of Multiscale Computational Methods in Polymeric Materials*. Polymers, 2017. **9** (1): p. 16.
19. Szabo, A. and N.S. Ostlund, *Modern Quantum Chemistry* . 1996, Mineola, New York: Dover Publications.
20. Parr, R.G. and W. Yang, *Density-functional theory of atoms and molecules* . International series of monographs on chemistry. 16. 1989, Oxford: Oxford University Press. ix, 333 p.
21. Karplus, M. and J. Kuriyan, *Molecular dynamics and protein function*. Proc Natl Acad Sci U S A, 2005. **102** (19): p. 6679-85.
22. Karplus, M., et al., *Molecular dynamics: applications to proteins*. Cold Spring Harb Symp Quant Biol, 1987. **52** : p. 381-90.
23. Berman, H.M., et al., *The Protein Data Bank*. Nucleic Acids Res., 2000. **28** : p. 235-242.
24. Glenn, J.M., J.T. Douglas, and L.K. Michael, *Constant pressure molecular dynamics algorithms*. The Journal of Chemical Physics, 1994.**101** (5): p. 4177-4189.
25. Brooks, B.R., et al., *Charmm - a Program for Macromolecular Energy, Minimization, and Dynamics Calculations*. Journal of Computational Chemistry, 1983. **4** (2): p. 187-217.
26. Weiner, P.W. and P.A. Kollman, *AMBER: assisted model building with energy refinement*. J. Comput. Chem., 1981. **2** : p. 287-303.
27. Scott, W.R.P., *The GROMOS biomolecular simulation program package*. J. Phys. Chem. A, 1999. **103** : p. 3596-3607.
28. Phillips, J.C., et al., *Scalable molecular dynamics with NAMD*. Journal of Computational Chemistry, 2005. **26** : p. 1781-1802.
29. Humphrey, W., A. Dalke, and K. Schulten, *VMD - Visual Molecular Dynamics*. Journal of Molecular Graphics, 1996. **14** : p. 33-38.
30. Allen, M.P. and D.J. Tildesley, *Computer simulation of liquids* . 1987, Oxford: Oxford science publications.
31. Caffarel, M. and P. Claverie, *Development of a pure diffusion quantum Monte Carlo method using a full generalized Feynman-Kac formula. I. Formalism*. The Journal of Chemical Physics, 1988.**88** (2): p. 1088-1099.
32. Gillespie, D.T., *Exact simulations of coupled chemical reactions*. J. Phys. Chem., 1977. **81** : p. 2340-2361.
33. Rotne, J. and S. Prager, *Variational Treatment of Hydrodynamic Interaction in Polymers*. The Journal of Chemical Physics, 1969.**50** (11): p. 4831-4837.
34. Yamakawa, H., *Transport Properties of Polymer Chains in Dilute Solution: Hydrodynamic Interaction*. The Journal of Chemical Physics, 1970. **53** (1): p. 436-443.

35. Brady, J.F. and G. Bossis, *Stokesian dynamics*. Ann. Rev. Fluid Mech., 1988. **20** : p. 111–157.
36. Noguchi, H., N. Kikuchi, and G. Gompper, *Particle-based mesoscale hydrodynamic techniques*. Europhys. Lett., 2007.**78** (1): p. 10005.
37. Español, P., *Hydrodynamics from dissipative particle dynamics*. Phys. Rev. E, 1995.
38. Groot, R.D. and P.B. Warren, *Dissipative particle dynamics: Bridging the gap between atomistic and mesoscopic simulation*. J. Chem. Phys., 1997. **107** (11): p. 4423.
39. Warren, P.B., *Dissipative particle dynamics*. Curr Opin Colloid in, 1998. **3** (6): p. 620–624.
40. Gao, L., J. Shillcock, and R. Lipowsky, *Improved dissipative particle dynamics simulations of lipid bilayers*. J. Chem. Phys., 2007.**126** (1): p. 015101.
41. Noguchi, H. and G. Gompper, *Transport coefficients of dissipative particle dynamics with finite time step*. Europhys. Lett., 2007. **79** : p. 36002.
42. Hu, H.H., N.A. Patankar, and M.Y. Zhu, *Direct Numerical Simulations of Fluid–Solid Systems Using the Arbitrary Lagrangian–Eulerian Technique*. J. Comp. Phys., 2001.**169** (2): p. 427–462.
43. Chen, S. and G.D. Doolen, *Lattice Boltzmann method for fluid flows*. Annual Review of Fluid Mechanics, 1998. **30** : p. 329–364.
44. Dupin, M.M., I. Halliday, and C.M. Care, *Multi-component lattice Boltzmann equation for mesoscale blood flow*. J. Phys. A, 2003.**36** : p. 8517.
45. Hauge, E.H. and A. Martin-Lñf, *Fluctuating hydrodynamics and Brownian motion*. Journal of Statistical Physics, 1973. **7** (3): p. 259–281.
46. Ramakrishnan, N., et al., *Motion of a nano-ellipsoid in a cylindrical vessel flow: Brownian and hydrodynamic interactions*.Journal of Fluid Mechanics, 2017. **821** : p. 117–152.
47. Asanovic, K., et al., *A view of the parallel computing landscape*. Commun. ACM, 2009. **52** (10): p. 56–67.
48. Dematté, L. and D. Prandi, *GPU computing for systems biology*.Briefings in Bioinformatics, 2010. **11** (3): p. 323–333.
49. Frenkel, D. and B. Smit, *Understanding molecular simulation: from algorithms to applications* . 2002, San Diego: Academic Press.
50. Heffelfinger, G.S. and M.E. Lewitt, *A comparison between two massively parallel algorithms for Monte Carlo computer simulation: An investigation in the grand canonical ensemble*. Journal of Computational Chemistry, 1996. **17** (2): p. 250–265.
51. Shimojo, F., et al., *Embedded divide-and-conquer algorithm on hierarchical real-space grids: parallel molecular dynamics simulation based on linear-scaling density functional theory*. Computer Physics Communications, 2005. **167** (3): p. 151–164.
52. McCammon, J.A. and S.C. Harvey, *Dynamics of Proteins and Nucleic Acids* . 1987, Cambridge, MA: Cambridge University Press.
53. McCammon, J.A., B.R. Gelin, and M. Karplus, *Dynamics of Folded Proteins*. Nature, 1977. **267** : p. 585–590.
54. Essmann, U., *A smooth particle mesh Ewald method*. J. Chem. Phys., 1995. **103** : p. 8577–8593.
55. Dongarra, J., S. Gottlieb, and W.T. Kramer, *Race to Exascale*.Computing in Science & Engineering, 2019. **21** (1): p. 4–5.
56. Gota, K., et al., *Application of MDGRAPE-3, a special purpose board for molecular dynamics simulations, to periodic biomolecular systems*. Journal of Computational Chemistry, 2009. **30** (1): p. 110–118.

57. Suenaga, A., et al., *Molecular Dynamics Simulations Reveal that Tyr-317 Phosphorylation Reduces Shc Binding Affinity for Phosphotyrosyl Residues of Epidermal Growth Factor Receptor*. 2009.**96** (6): p. 2278-2288.
58. Shaw, D.E., et al., *Anton, a special-purpose machine for molecular dynamics simulation*. Communications of the Acm, 2008.**51** (7): p. 91-97.
59. Shan, Y., et al., *A conserved protonation-dependent switch controls drug binding in the Abl kinase*. Proc Natl Acad Sci U S A, 2009. **106** (1): p. 139-44.
60. Friedrichs, M.S., et al., *Accelerating molecular dynamic simulation on graphics processing units*. J Comput Chem, 2009.**30** (6): p. 864-72.
61. Stone, J.E., et al., *Accelerating molecular modeling applications with graphics processors*. J Comput Chem, 2007.**28** (16): p. 2618-40.
62. Redondo, A. and R. LeSar, *MODELING AND SIMULATION OF BIOMATERIALS*. Annual Review of Materials Research, 2004.**34** (1): p. 279-314.
63. Giuseppina, R. and G. Fabio, *Understanding the Performance of Biomaterials through Molecular Modeling: Crossing the Bridge between their Intrinsic Properties and the Surface Adsorption of Proteins*. Macromolecular Bioscience, 2007. **7** (5): p. 552-566.
64. E, W.N. and B. Engquist, *Multiscale Modeling in Computation*. Notices of the AMS, 2003. **50** (9): p. 1062-1070.
65. Dama, J.F., M. Parrinello, and G.A. Voth, *Well-Tempered Metadynamics Converges Asymptotically*. Physical Review Letters, 2014.**112** (24): p. 240602.
66. Bolhuis, P.G., et al., *Transition path sampling: Throwing ropes over rough mountain passes, in the dark*. Annual Review of Physical Chemistry, 2002. **53** : p. 291-318.
67. Watanabe, M. and M. Karplus, *Simulations of Macromolecules by Multiple Time-Step Methods*. J. Phys. Chem., 1995. **99** : p. 5680-5697.
68. Press, W.H., et al., *Numerical recipes in C (2nd ed.): the art of scientific computing* . 1992: Cambridge University Press.
69. Kevrekidis, I.G., et al., *Equation-free, coarse-grained multiscale computation: enabling microscopic simulators to perform system-level analysis*. Communications in Mathematical Sciences, 2003.**1** : p. 715-762.
70. Bindal, A., et al., *Equation-free, coarse-grained computational optimization using timesteppers*. Chemical Engineering Science, 2006. **61** (2): p. 779-793.
71. Bradley, R. and R. Radhakrishnan, *Coarse-Grained Models for Protein-Cell Membrane Interactions*. Polymers, 2013. **5** (3): p. 890-936.
72. Li, Y., et al., *Challenges in Multiscale Modeling of Polymer Dynamics*. Polymers, 2013. **5** (2): p. 751-832.
73. Klein, M.L. and W. Shinoda, *Large-scale molecular dynamics simulations of self-assembling systems*. Science, 2008.**321** (5890): p. 798-800.
74. Ayton, G.S., E. Lyman, and G.A. Voth, *Hierarchical coarse-graining strategy for protein-membrane systems to access mesoscopic scales*. Faraday Discuss., 2010. **144** : p. 347.
75. Marrink, S.J., et al., *The MARTINI force field: coarse grained model for biomolecular simulations*. J Phys Chem B, 2007.**111** (27): p. 7812-24.
76. Ahmadi, S., et al., *Multiscale modeling of enzymes: QM-cluster, QM/MM, and QM/MM/MD: A tutorial review*. International Journal of Quantum Chemistry, 2018. **118** (9): p. e25558.

77. Shi, Q., S. Izvekov, and G.A. Voth, *Mixed Atomistic and Coarse-Grained Molecular Dynamics: Simulation of a Membrane-Bound Ion Channel*. The Journal of Physical Chemistry B, 2006. **110** (31): p. 15045-15048.
78. Yasuda, S. and R. Yamamoto, *Multiscale modeling and simulation for polymer melt flows between parallel plates*. Physical Review e, 2010. **81** (3).
79. Ramakrishnan, N., et al., *Biophysics of membrane curvature remodeling at molecular and mesoscopic lengthscales*. J Phys Condens Matter, 2018. **30** (27): p. 273001.
80. Borgdorff, J., et al., *Distributed multiscale computing with MUSCLE 2, the Multiscale Coupling Library and Environment*. Journal of Computational Science, 2014. **5** (5): p. 719-731.
81. Wolstencroft, K., et al., *The Taverna workflow suite: designing and executing workflows of Web Services on the desktop, web or in the cloud*. Nucleic Acids Res, 2013. **41** (Web Server issue): p. W557-61.
82. Jabeen, Z., et al., *Rheology of colloidal suspensions in confined flow: Treatment of hydrodynamic interactions in particle-based simulations inspired by dynamical density functional theory*. Physical Review E, 2018. **In Press** .
83. Yu, H.-Y., et al., *Microstructure of Flow-Driven Suspension of Hardspheres in Cylindrical Confinement: A Dynamical Density Functional Theory and Monte Carlo Study*. Langmuir, 2017. **33** (42): p. 11332-11344.
84. Fredrikson, G., *The Equilibrium Theory of Inhomogeneous Polymers* . 2006: Oxford University Press.
85. Ravi Radhakrishnan, H.-Y.Y., David M. Eckmann, Portovovo S. Ayyaswamy, *Computational Models for Nanoscale Fluid Dynamics and Transport Inspired by Nonequilibrium Thermodynamics*. J Heat Transfer, 2017. **39** : p. 033001.
86. Zwanzig, R.W., *High-Temperature Equation of State by a Perturbation Method. I. Nonpolar Gases*. The Journal of Chemical Physics, 1954. **22** (8): p. 1420-1426.
87. Beveridge, D.L. and F.M. DiCapua, *Free energy via molecular simulation: applications to chemical and biomolecular systems*. Annu Rev Biophys Biophys Chem, 1989. **18** : p. 431-92.
88. Chandler, D., *Statistical mechanics of isomerization dynamics in liquids and the transition state approximation*. J. Chem. Phys., 1978. **68** : p. 2959-2970.
89. Bartels, C. and M. Karplus, *Probability distribution for complex systems: Adaptive umbrella sampling of the potential energy*. J. Phys. Chem. B, 1998. **102** : p. 865-880.
90. Roux, B., *The Calculation of the Potential of Mean Force Using Computer-Simulations*. Computer Physics Communications, 1995.**91** (1-3): p. 275-282.
91. Barducci, A., G. Bussi, and M. Parrinello, *Well-Tempered Metadynamics: A Smoothly Converging and Tunable Free-Energy Method*. Physical Review Letters, 2008. **100** (2): p. 020603.
92. Weinan, E., W.Q. Ren, and E. Vanden-Eijnden, *Transition pathways in complex systems: Reaction coordinates, isocommittor surfaces, and transition tubes*. Chemical Physics Letters, 2005.**413** (1-3): p. 242-247.
93. Elber, R., A. Ghosh, and A. Cardenas, *Long time dynamics of complex systems*. Acc. Chem. Res., 2002. **35** : p. 396-403.
94. Elber, R., J. Meller, and R. Olender, *Stochastic path approach to compute atomically detailed trajectories: Application to the folding of C peptide*. J. Phys. Chem. B, 1999. **103** : p. 899-911.
95. Elber, R., et al., *Bridging the gap between long time trajectories and reaction pathways*. Adv. Chem. Phys, 2003.**126** : p. 93-129.
96. Henkelman, G. and H. Jonsson, *Improved tangent estimate in the nudged elastic band method for finding minimum energy paths and saddle points*. J. Chem. Phys., 2000. **113** : p. 9978.

97. Henkelman, G., B.P. Uberuaga, and H. Jonsson, *A climbing image nudged elastic band method for finding saddle points and minimum energy paths*. J. Chem. Phys., 2000. **113** : p. 9901.
98. Jonsson, H., G. Mills, and K.W. Jacobsen, *Nudged elastic band method for finding minimum energy paths of transitions* . Classical and quantum dynamics in condensed phase simulations. 1998: World Scientific.
99. E, W.N., W. Ren, and E. Vanden-Eijnden, *Finite temperature string method for the study of rare events*. J. Phys. Chem. B, 2005.**109** : p. 6688-6693.
100. Dellago, C., P.G. Bolhuis, and P.L. Geissler, *Transition Path Sampling*. Adv. Chem. Phys, 2002. **123** : p. 1-81.
101. Bolhuis, P.G., C. Dellago, and D. Chandler, *Sampling ensembles of deterministic transition pathways*. Faraday Discuss., 1998.**110** : p. 421-436.
102. Nielsen, S.O. and M.L. Klein, *A coarse grain model for lipid monolayer and bilayer studies*. Lecture notes in physics, 2002.**605** : p. 27-63.
103. Izvekov, S. and G.A. Voth, *Multiscale coarse graining of liquid-state systems*. Journal of Chemical Physics, 2005.**123** (13): p. -.
104. Izvekov, S. and G.A. Voth, *Multiscale coarse-graining method for biomolecular systems*. J. Phys. Chem. B, 2005. **109** : p. 2469-2473.
105. Izvekov, S., et al., *Effective force fields for condensed phase systems from ab initio molecular dynamics simulation: A new method for force-matching*. Journal of Chemical Physics, 2004.**120** (23): p. 10896-10913.
106. Periole, X., et al., *Combining an Elastic Network With a Coarse-Grained Molecular Force Field: Structure, Dynamics, and Intermolecular Recognition*. Journal of Chemical Theory and Computation, 2009. **5** (9): p. 2531-2543.
107. Jensen, F., *Introduction to Computational Chemistry* . 2nd ed. 2007, Chichester: John Wiley & sons.
108. Warshel, A. and W.W. Parson, *Dynamics of Biochemical and Biophysical Reactions: Insight from Computer Simulations*. Quart. Rev. Biophys., 2001. **34** : p. 563-679.
109. Warshel, A., *Computer modeling of chemical reactions in enzymes and solution* . 1989, New York: John Wiley and Sons.
110. Shurki, A. and A. Warshel, *Structure/function correlations of proteins using MM, QM/MM, and related approaches: Methods, concepts, pitfalls, and current progress*. Protein Simulations, 2003. **66** : p. 249-313.
111. Senn, H.M. and W. Thiel, *QM/MM methods for biological systems* , in *Atomistic Approaches in Modern Biology: from Quantum Chemistry to Molecular Simulations* . 2007, Springer-Verlag Berlin: Berlin. p. 173-290.
112. Das, D., et al., *Optimization of quantum mechanical molecular mechanical partitioning schemes: Gaussian delocalization of molecular mechanical charges and the double link atom method*. Journal of Chemical Physics, 2002. **117** (23): p. 10534-10547.
113. Reuter, N., et al., *Frontier bonds in QM/MM methods: A comparison of different approaches*. Journal of Physical Chemistry A, 2000. **104** (8): p. 1720-1735.
114. Field, M.J., P.A. Bash, and M. Karplus, *A combined quantum mechanical and molecular mechanical potential for molecular dynamics simulations*. J. Comput. Chem., 2002. **11** : p. 700-733.
115. Zhang, Y. and W. Yang, *A pseudobond approach to combining quantum mechanical and molecular mechanical methods*. J. Chem. Phys., 1999. **110** : p. 46-54.

116. Garcia-Viloca, M. and J. Gao, *Generalized Hybrid Orbital for the treatment of boundary atoms in combined quantum mechanical and molecular mechanical calculations using the semiempirical parameterized model 3 method*. Theoretical Chemistry Accounts, 2004. **111** : p. 280-286.
117. Pu, J., D.G. Truhlar, and J. Gao, *The Generalized Hybrid Orbital (GHO) method for ab initio combined QM/MM calculations*. Journal of Physical Chemistry A, 2004. **108** : p. 632-650.
118. Friesner, R.A., et al., *How iron-containing proteins control dioxygen chemistry: a detailed atomic level description via accurate quantum chemical and mixed quantum mechanics/molecular mechanics calculations*. Coordination Chemistry Reviews, 2003. **238-239** : p. 267-290.
119. Rega, N., et al., *Hybrid ab initio empirical molecular dynamics: combining the ONIOM scheme with the atom-centered density matrix propagation (ADMP) approach*. J. Phys. Chem. B, 2004.**108** : p. 4210-4220.
120. Mulholland, A.J., *Modelling enzyme reaction mechanisms, specificity and catalysis*. Drug Discovery Today, 2005. **10** (20): p. 1393-1402.
121. Schmidt, M.W., et al., *General Atomic and Molecular Electronic-Structure System*. Journal of Computational Chemistry, 1993.**14** (11): p. 1347-1363.
122. Zhou, R. and B.J. Berne, *A New Molecular Dynamics Method Combining the Reference System Propagator Algorithm with a Fast Multipole Method for Simulating Proteins and Other Complex Systems*. J. Chem. Phys., 1995. **103** : p. 9444-9459.
123. Car, R. and M. Parrinello, *Unified Approach for Molecular-Dynamics and Density-Functional Theory*. Physical Review Letters, 1985. **55** (22): p. 2471-2474.
124. Galli G. and P. A., *First Principles Molecular Dynamics* . Computer Simulation in Chemical Physics, NATO ASI Series (Series C: Mathematical and Physical Sciences), ed. M.P. Allen and D.J. Tildesley. Vol. 397. 1993, Dordrecht: Springer.
125. Park, J.H. and A. Heyden, *Solving the equations of motion for mixed atomistic and coarse-grained systems*. Molecular Simulation, 2009.**35** (10-11): p. 962-973.
126. Wijesinghe, H.S. and N.G. Hadjiconstantinou, *A hybrid atomistic-continuum formulation for unsteady, viscous, incompressible flows*. Cmes-Computer Modeling in Engineering & Sciences, 2004.**5** (6): p. 515-526.
127. Hadjiconstantinou, N.G., *Hybrid atomistic-continuum formulations and the moving contact-line problem*. Journal of Computational Physics, 1999. **154** : p. 245-265.
128. Vitoshkin, H., et al., *Nanoparticle stochastic motion in the inertial regime and hydrodynamic interactions close to a cylindrical wall*. Phys Rev Fluids, 2016. **1** .
129. Hsiu-Yu Yu, D.M.E., Portovovo S. Ayyaswamy, Ravi Radhakrishnan, *Effect of wall-mediated hydrodynamic fluctuations on the kinetics of a Brownian nanoparticle*. Proceedings of the Royal Society of London. Series A, 2016. **472** : p. 20160397.
130. Yu, H.-Y., et al., *Composite generalized Langevin equation for Brownian motion in different hydrodynamic and adhesion regimes*. Physical Review E, 2015. **91** (5): p. 052303.
131. Farokhirad, S., et al., *Stiffness can mediate balance between hydrodynamic forces and avidity to impact the targeting of flexible polymeric nanoparticles in flow*. Nanoscale, 2019. **11** (14): p. 6916-6928.
132. Lee, M.J.a.A., S Ye and Gardino, Alexandra K and Heijink, Anne Margriet and Sorger, Peter K and MacBeath, Gavin and Yaffe, Michael B, *Sequential application of anticancer drugs enhances cell death by rewiring apoptotic signaling networks*. Cell, 2012. **149** (4): p. 780-794.
133. Janes, K.A., et al., *A systems model of signaling identifies a molecular basis set for cytokine-induced apoptosis*. Science, 2005.**310** (5754): p. 1646-53.

134. Janes, K.A., et al., *A high-throughput quantitative multiplex kinase assay for monitoring information flow in signaling networks: application to sepsis-apoptosis*. Mol Cell Proteomics, 2003.**2** (7): p. 463-73.
135. Sachs, K., et al., *Causal Protein-Signaling Networks Derived from Multiparameter Single-Cell Data*. Science, 2005.**308** (5721): p. 523.
136. Jordan, E.J., et al., *Computational algorithms for in silico profiling of activating mutations in cancer*. Cell Mol Life Sci, 2019.**76** (14): p. 2663-2679.
137. Agajanian, S., O. Oluyemi, and G.M. Verkhivker, *Integration of Random Forest Classifiers and Deep Convolutional Neural Networks for Classification and Biomolecular Modeling of Cancer Driver Mutations*.Frontiers in Molecular Biosciences, 2019. **6** : p. 44.
138. Peskin, C.S. and D.M. McQueen, *A 3-dimensional computational method for blood-flow in the heart .1. Immersed elastic fibers in a viscous incompressible fluid*. Journal of Computational Physics, 1989.**81** (2): p. 372-405.
139. Wu, X.L., et al., *Adaptive nonlinear finite elements for deformable body simulation using dynamic progressive meshes*. Computer Graphics Forum, 2001. **20** (3): p. C349-C358.
140. Tompson, J., et al., *Accelerating Eulerian Fluid Simulation With Convolutional Networks* , in *Proceedings of the 34 th International Conference on Machine Learning* . 2017: Sydney, Australia.
141. Balaprakash, P., et al., *Autotuning in High-Performance Computing Applications*. Proceedings of the IEEE, 2018.**106** (11): p. 2068-2083.
142. Jordan, M.I. and T.M. Mitchell, *Machine learning: Trends, perspectives, and prospects*. Science, 2015. **349** (6245): p. 255-60.
143. Carlson, J.M. and J. Doyle, *Complexity and robustness*. Proc Natl Acad Sci U S A, 2002. **99 Suppl 1** : p. 2538-45.
144. Gelman, A., et al., *Bayesian data analysis* . 2013: Chapman Hall.
145. Alber, M., et al., *Integrating machine learning and multiscale modeling-perspectives, challenges, and opportunities in the biological, biomedical, and behavioral sciences*. NPJ Digit Med, 2019.**2** : p. 115.
146. runton, S., B. Noack, and P. Koumoutsakos, *Machine learning for fluid mechanics*. arXiv preprint arXiv:1905.11075, 2019.
147. Lillicrap, T.P. and K.P. Kording, *What does it mean to understand a neural network?* arXiv:1907.06374, 2019.
148. Raissi, M., P. Perdikaris, and G.E. Karniadakis, *Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations*. Journal of Computational Physics, 2019.**378** : p. 686-707.
149. Hornik, K., M. Stinchcombe, and H. White, *Multilayer feedforward networks are universal approximators*. Neural Networks, 1989. **2** (5): p. 359-366.
150. Baydin, A.G., et al., *Automatic Differentiation in Machine Learning: a Survey*. Journal of Machine Learning, 2018.**18** (153): p. 1-43.
151. Lachmann, A., et al., *ARACNe-AP: gene network reverse engineering through adaptive partitioning inference of mutual information*. Bioinformatics, 2016. **32** (14): p. 2233-5.
152. Margolin, A.A., et al., *ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context*. BMC Bioinformatics, 2006. **7 Suppl 1** : p. S7.

153. Cover, T.M., *Elements of information theory* . 2012: John Wiley and Sons.
154. Watanabe, S., *Algebraic geometry and statistical learning theory* . Vol. 25. 2009: Cambridge University Press.
155. Amari, S.-i., *Information geometry and its applications* . Vol. 194. 2016: Springer.
156. Pratik Chaudhari, A.C., Stefano Soatto, Yann LeCun, Carlo Baldassi, Christian Borgs, Jennifer Chayes, Levent and a.R.Z. Sagun, *Entropy-SGD: biasing gradient descent into wide valleys*. In Proc. of International Conference of Learning and Representations,, 2016.
157. Pratik Chaudhari, A.O., Stanley Osher, Stefano Soatto, and Carlier Guillaume, *Deep Relaxation: partial differential equations for optimizing deep neural networks*. Research in the Mathematical Sciences arXiv:1704.04932, 2018.
158. Pratik Chaudhari, C.B., Riccardo Zecchina, Stefano Soatto, Ameet Talwalkar, and Adam Oberman, *Parle: parallelizing stochastic gradient descent*. arXiv:1707.00424, 2017.
159. Chmiela, S., et al., *Towards exact molecular dynamics simulations with machine-learned force fields*. Nature Communications, 2018. **9** (1): p. 3887.
160. Batra, R. and S. Sankaranarayanan, *Machine learning for multi-fidelity scale bridging and dynamical simulations of materials*. Journal of Physics: Materials, 2020. **3** (3): p. 031002.
161. Tuckerman, M.E., *Machine learning transforms how microstates are sampled*. Science, 2019. **365** (6457): p. 982.
162. Rogal, J., E. Schneider, and M.E. Tuckerman, *Neural-Network-Based Path Collective Variables for Enhanced Sampling of Phase Transformations*. Phys Rev Lett, 2019.**123** (24): p. 245701.
163. Chiavazzo, E., et al., *Intrinsic map dynamics exploration for uncharted effective free-energy landscapes*. Proc Natl Acad Sci U S A, 2017. **114** (28): p. E5494-E5503.
164. Zhang, J. and M. Chen, *Unfolding Hidden Barriers by Active Enhanced Sampling*. Phys Rev Lett, 2018. **121** (1): p. 010601.
165. Sultan, M.M. and V.S. Pande, *Automated design of collective variables using supervised machine learning*. The Journal of Chemical Physics, 2018. **149** (9): p. 094106.
166. Alom, Z.M., et al., *A State-of-the-Art Survey on Deep Learning Theory and Architectures*. Electronics, 2019. **8** (3).
167. Bonati, L., Y.-Y. Zhang, and M. Parrinello, *Neural networks-based variationally enhanced sampling*. Proceedings of the National Academy of Sciences, 2019. **116** (36): p. 17641.
168. Radhakrishnan, R. and T. Schlick, *Biomolecular free energy profiles by a shooting/umbrella sampling protocol, "BOLAS"*. J Chem Phys, 2004. **121** (5): p. 2436-44.
169. Ma, H., M. Govoni, and G. Galli, *Quantum simulations of materials on near-term quantum computers*. Cond. mat. Arxiv, 2020: p. arXiv:2002.11173v1.
170. Sieck, G.C., *Physiology in Perspective: Physiology is Everywhere*. Physiology (Bethesda), 2019. **34** (3): p. 167-168.
171. Ellis, G.F.R. and J. Kopel, *The Dynamical Emergence of Biology From Physics: Branching Causation via Biomolecules*. Front Physiol, 2018. **9** : p. 1966.