MULTI-SCALE METHODS IN TIME AND SPACE FOR PARTICLE SIMULATIONS

A DISSERTATION
SUBMITTED TO THE INSTITUTE FOR COMPUTATIONAL
AND MATHEMATICAL ENGINEERING
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

William Fong

July 2009

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Eric Darve) Principal Adviser

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Adrian Lew)

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Wei Cai)

Approved for the University Committee on Graduate Studies.

Abstract

To be added later

Acknowledgements

I would like to thank the reading committee members for attempting to make it through this looooooooong manuscript.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

Particle simulations arises in many applications such as gravitational N-body problems, electrostatics, and molecular dynamics simulations. All of these problems involve a system of particles, each with a mass and a position, that interact according to a potential function. The motion of the particles is governed by Newton's equations of motion. Since the potential is typically non-linear and couples all of the particles, this system of ordinary differential equations cannot be solved analytically, except for a few special cases. Therefore the solution requires numerical simulation via time integration. Given an initial condition, time integration advances the system of particles forward in time with a discrete time step. Each integration step involves two parts: (1) first the force on each particle is computed using the gradient of the potential function; (2) then the positions of the particles are advanced forward one time step using the computed force. These two parts are repeated until the system arrives at the desired final time. Of the two, the force evaluation step is usually more expensive since the presence of long-range forces such as gravity or electrostatics requires an all-to-all computation i.e. each particle interacts with every other particle. For an N-particle system, the direct evaluation of the force is an $O(N^2)$ procedure which quickly becomes intractable for large systems. Two common approaches for improving the computational efficiency of the force evaluation, and hence particle simulations, are computing the force less frequently and computing the force more efficiently.

One way to compute the force less often is to assign a larger time step to those components of the potential function that correspond to long-range forces which are expensive to evaluate. This idea of multiple-time-stepping (MTS) have been explored in detail by previous work such as [add list of citations]. In the context of particle simulations, a popular MTS scheme is r-RESPA [90, 38] where each pair of time steps is required to be in integer ratio. If the use of several time steps is desired, this condition limits the flexibility in the choice of time steps. A class of integrators that allows for the time steps to be in arbitrary ratio is asynchronous variational integrators (AVI) [59, 58, 57]. However the formulation and application of AVI in the cited work is restricted to finite-element simulations. In this dissertation, we will adapt AVI for particle simulations in order to remove the time step restriction of r-RESPA and show that AVI is indeed a generalization of r-RESPA. Although MTS schemes can be used to reduce the cost of particle simulations, there are potential drawbacks. One such example is the presence of instabilities whose manifestation is dependent on the choice of time steps [add list of citations]. Since r-RESPA is a special instance of AVI, we expect AVI to inherit the instabilities observed in the application of r-RESPA. To verify this, a thorough analysis of the stability of AVI is also provided in the thesis.

While using multiple time steps reduces the cost of force evaluations, this reduction is only by a constant factor i.e. the scaling is still $O(N^2)$ for long-range forces. Perhaps a more pragmatic approach is to develop better scaling methods for evaulating the force. Towards this end there are two main classes of methods, the particle mesh Ewald (PME) method and the fast multipole method (FMM). PME [add citations] is a grid-based $O(N \log N)$ method that utilizes fast Fourier transforms (FFTs) to compute $1/r$ interactions (e.g. gravitation, electrostatics) for periodic systems. Despite being quite popular for molecular dynamics simulations, PME will not be discussed any further in this dissertation. On the other hand, the FMM [add citations] is a tree-based $O(N)$ method that uses analytical formulae to construct a low-rank approximation to represent far-field $1/r$ interactions. Since the low-rank approximation is dependent on the functional form of the kernel $1/r$, a new FMM would need to be derived and implemented for other types of interactions. Therefore it would be convenient to have a black-box method that is applicable to a wide class of kernels and would simply require the user to provide a routine that evaluates the kernel. The construction of a black-box FMM comprises the next part of the dissertation.

Besides computational efficiency, improvements can also be made by introducing novel techniques that enable the use of particle simulations in a new application area. An example of such an area is the study of the mechanical properties of nanowires, a cylindrical crystalline structure whose size is on the nano-scale. In particular it is difficult to study the torsion and bending of nanowires experimentally due to their small size. Even if the nanowire can be clamped down and held in place for tests to be conducted, the end effects can adversely affect the data [add citations]. Another approach for this study is through numerical simulation. To remove the end effects, periodic boundary conditions (PBCs) can be used to simulate an infinitely long nanowire. However the current formulation of PBCs does not allow for the application of torque or a bend. In the final part of this dissertation, we will discuss how PBCs can be reformulated to handle torsion and bending and illustrate how this technique can be applied to the molecular dynamics simulations of silicon nanowires.

1.2 Outline of Dissertation

This dissertation consists of three parts: (1) multi-scale methods in time (2) multi-scale methods in space and (3) application of particle simulations to study mechanical properties of nanowires.

For the work on multi-scale methods in time, we start in Chapter 2 with an introduction to time integrators from a variational perspective and review how to derive single-time-step integrators in this context. After discussing how the Trotter factorization can alternatively be used to construct single-time-step integrators, the process of building r-RESPA from this framework is summarized. We finish the chapter by detailing how AVI can be constructed for particle simulations. In Chapter 3 we investigate the stability of AVI in particle simulations. To start we focus on a two-spring single mass system, a simple model which results from linearizing the potential function around an minimum energy configuration. Assuming that one spring is stiff while the other is soft, a different time step is selected for each spring and the stability of the time integration for various time steps is examined. A rigorous linear stability analysis is performed in addition to a host of empirical studies. Next the analysis is extended to a 2-D periodic harmonic lattice, a more complex model that better mimics the systems typically encountered in particle simulations. Finally we look into the differences in the stability of AVI in molecular dynamics versus finite element simulations. Due to the presence of resonances when large time steps are taken in molecular dynamics simulations, Chapter 4 explains how AVI can be extended for Langevin dynamics, a stochastic thermostat used for damping these instabilities. After showing how an analog of AVI in the Langevin setting is constructed, we finish with a numerical simulation of a peptide using the stochastic integrator.

We then shift our attention to multi-scale methods in space. In Chapter 5 a review of the classical FMM is provided. Namely we start by detailing how a low-rank approximation coupled with a proper spatial decomposition can be used to build a fast summation method. We conclude with a summary of the analytical formulae needed to achieve the $O(N)$ scaling of the FMM. Next in Chapter 6 a black-box approach is introduced that is applicable to a wide class of kernels, including the $1/r$ kernel at the heart of the classical FMM. We start by showing how a low-rank approximation can be constructed using Chebyshev interpolation. This approximation can be viewed as a black box since it is independent of the functional form of the kernel. Pairing the low-rank approximation with the FMM machinery gives the black-box FMM (bbFMM). Next we discuss a strategy for accelerating the method through the use of the singular value decomposition to reduce the size of the matrix-vector products involved. Numerical results obtained from the application of the method to various kernels wrap up the chapter. A discussion on the application of bbFMM to periodic systems is provided in Chapter 7. After showing how to extend bbFMM to absolutely convergent periodic sums, we explain that some extra handling is needed for periodic sums with only conditional convergence. To close, numerical results illustrating this procedure are given.

Finally in Chapter 8 we show how to formulate periodic boundary conditions for molecular dynamics simulations on the torsion and bending of silicon nanowires. This allows us to use simulations to investigate the mechanical failure of nanowires from torsion and bending without undesirable end effects. Derivations of the expressions for the virial torque and virial bending moment are provided. Such an expression gives the user a means of measuring the amount of torque (or bending moment) being applied to the nanowire under torsion (or bending) PBCs. Various failure phenomena are observed and presented at the end of the chapter.

Chapter 2

Variational Integrators

Symplectic integrators are usually adopted as the integrators of choice for simulation of systems of particles (bio-molecules, proteins, crystals, gravitational N-body problems, ...) and for some computational mechanics applications. One of the reasons behind this choice is their excellent long-time energy conservation properties, which can be traced back to the existence of a shadow Hamiltonian function almost exactly conserved by the numerical trajectory, see, e.g. [78, 8].

A powerful and flexible approach for deriving symplectic integrators stems from a discrete version of Hamilton's principle, which led to the development of variational integrators [87, 61, 66, 58]. In this approach the starting point consists in constructing a suitable approximation to the action integral, termed the action sum. The algorithm then follows by requesting the discrete trajectory to be a stationary point of the action sum. Any variational integrator is symplectic, and conversely. Additionally, variational integrators can be constructed to have other outstanding conservation properties by taking advantage of a discrete version of Noether's theorem [60, 66, 59], which guarantees that for each symmetry operation of the discrete action there exists a corresponding conserved quantity, as in the continuous-time case. These and other features of the theory of variational integrators have been thoroughly discussed in many earlier references (see previous references and [93, 65, 92, 70, 58, 66]); hence we shall skip further discussions herein.

Asynchronous variational integrators (AVI) are a class of variational integrators distinguished by the trait of enabling the use of different time steps for different potential energy contributions to a mechanical system. Their formulation and use in the context of finite element (FE) discretizations in solids and some fluid mechanics simulations can be found in [59, 58, 57], and they essentially amount to considering a possibly-different time step for each element in the mesh. These algorithms share many features with other multiple time step methods in computational mechanics, commonly known as subcycling or element-by-element methods [73, 7, 7, 73, 47, 83, 22, 20, 19, 35, 34]. In the context of particle simulations, in particular molecular dynamics, r-RESPA [90, 38] is perhaps the most widely known multiple time step method. Here different potential energy terms are integrated with time steps in integer ratios. In this chapter we will show how AVI can be formulated for particle simulations by generalizing r-RESPA to arbitrary time step ratios.

We begin by describing the variational principle in the continuous-time setting. We then extend this framework to the discrete case and show how variational integrators can be obtained. As an example this procedure will be demonstrated for the velocity Verlet (VV) integrator. Next we present an alternative approach to deriving the VV integrator by using the Trotter factorization and apply these ideas recursively to obtain r-RESPA. Finally we close by including the derivation of AVI from a discrete version of the variational principle and comment about its implementation. Although the algorithm is essentially identical to that introduced in [58], it is presented here in a framework better suited for particle simulations. The key difference is the definition of the positions for *all* the degrees of freedom at every potential update. This enables the construction of a single discrete Lagrangian between any two consecutive potential updates, as opposed to the discrete Lagrangians per element that naturally appear in the continuum mechanics setting of [58], but that do not have a natural analog in the ordinary differential equation case. By construction, it is then evident that AVI is symplectic. The AVI algorithm is presented in the non-staggered form which is commonly found in molecular dynamics integrators such as r-RESPA.

2.1 Continuous Variational Principle

The Lagrangian for a system of N particles is given by the difference between the kinetic and potential energies:

$$L(q, \dot{q}) = \sum_{a=1}^{N} \frac{1}{2} m_a \left|\left|\dot{q}_a\right|\right|^2 - V(q)$$

where m_a is the mass of particle a, q_a is the position of particle a, $q = \{q_a\}$ is the collection of all particle positions, $\dot{q}_a$ is the velocity of particle a, and $V(q)$ is a given potential function. The action of an arbitrary trajectory $q(t)$ of the Lagrangian system is defined as the time integral of $L(q(t), \dot{q}(t))$ over the interval of interest $[0, T]$:

$$S[q(\cdot)] = \int_0^T L(q(t), \dot{q}(t))dt.$$

Hamilton's variational principle states that the trajectory $q(t)$ followed by the particles is a stationary point of the action integral S among all smooth trajectories with the same initial and end points, $q(0)$ and $q(T)$, respectively, i.e., $\delta S = 0$ for any variation $\delta q(t)$ satisifying $\delta q(0) = 0$ and $\delta q(T) = 0$. Assuming that the trajectory $q(t)$ is a stationary point gives

$$0 = \delta S = \int_0^T \left(\frac{\partial L}{\partial q} \delta q + \frac{\partial L}{\partial \dot{q}} \delta \dot{q} \right) dt.$$

Using integration by parts and applying the boundary conditions $\delta q(0) = 0$ and $\delta q(T) = 0$ we have

$$0 = \left[\frac{\partial L}{\partial \dot{q}} \delta q \right]_{t=0}^{t=T} + \int_0^T \left(\frac{\partial L}{\partial q} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \right) \right) \delta q \, dt = \int_0^T \left(\frac{\partial L}{\partial q} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \right) \right) \delta q \, dt.$$

Since this result must hold for all variations $\delta q(t)$ we obtain the Euler-Lagrange equations for this variational principle:

$$\frac{\partial L}{\partial q} = \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \right).$$

Substituting in the Lagrangian defined above we have

$$-\frac{\partial V}{\partial q} = m\ddot{q}$$

which is precisely Newton's equations of motion.

2.2 Discrete Variational Principle

Variational integrators are constructed by mimicking this variational structure in the discrete case. The essential idea is to approximate the action over a discrete trajectory and use a discrete variational principle to obtain the algorithm. More precisely, the time interval $[0, T]$ is partitioned into a sequence of times $\{t^j\} = \{t^0 = 0, \ldots, t^M = T\}$ with a time step h, and a discrete trajectory on this partition is a sequence of positions $\{q^j\} = \{q^0, \ldots, q^M\}$. The approximation of the action, the action sum, is constructed as

$$S_{\mathrm{d}}[\{q^j\}] = \sum_{j=0}^{M-1} L_{\mathrm{d}}(q^j, q^{j+1}),$$

where $L_{\mathrm{d}}(q, \tilde{q})$ is the discrete Lagrangian. The integrator follows by employing a discrete version of Hamilton's principle: the discrete trajectory renders $\delta S_{\mathrm{d}} = 0$ for any variation δq satisfying $\delta q^0 = 0$ and $\delta q^M = 0$. The discrete Euler-Lagrange equations for this variational principle are

$$\frac{\partial}{\partial q} L_{\mathrm{d}}(q^j, q^{j+1}) + \frac{\partial}{\partial \tilde{q}} L_{\mathrm{d}}(q^{j-1}, q^j) = 0, \tag{2.1}$$

for $j = 1, \ldots, M-1$, where $\partial L_{\mathrm{d}}(\cdot, \cdot)/\partial q$ and $\partial L_{\mathrm{d}}(\cdot, \cdot)/\partial \tilde{q}$ indicate the partial derivatives of L_{d} with respect to its first and second arguments, respectively. The discrete momenta $\{p^j\}$ are generally defined, via a discrete Legendre transform, to be

$$p^j = \frac{\partial}{\partial \tilde{q}} L_{\mathrm{d}}(q^{j-1}, q^j) = -\frac{\partial}{\partial q} L_{\mathrm{d}}(q^j, q^{j+1}), \tag{2.2}$$

where the second equality follows from the discrete Euler-Lagrange equations (2.1). Eq. (2.2) implicitly defines a symplectic map $(q^j, p^j) \mapsto (q^{j+1}, p^{j+1})$, see e.g. [41], so all variational integrators are symplectic. As a result, variational integrators conserve energy very well (no drift) over long time scales [76, 39]. This is a key property for molecular dynamics and other problems.

To illustrate the procedure for deriving variational integrators we will show that by approximating the action with a particular discrete Lagrangian, the velocity Verlet (VV) integrator can be obtained, as shown for example in [53]. We start by recalling the VV integrator for a system of N particles with masses $m = (m_1, \ldots, m_N)$. Let the positions be given by Cartesian coordinates $x = (x_1, \ldots, x_N)$ with corresponding velocities $v = \dot{x} = (v_1, \ldots, v_N)$ and assume these particles interact according to a given potential function $V(x)$. Their trajectories $x(t)$ are governed by Newton's equations of motion $F = \mathbf{M}\, a$ where $F = -\nabla V(x)$, $\mathbf{M}$ is the diagonal matrix with the particle masses along the diagonal, and $a = \ddot{x}$. For this system the VV integrator is given by:

$$v^{j+1/2} = v^j + \frac{h}{2}\mathbf{M}^{-1}F^j \tag{2.3a}$$

$$x^{j+1} = x^j + h\, v^{j+1/2} \tag{2.3b}$$

$$v^{j+1} = v^{j+1/2} + \frac{h}{2}\mathbf{M}^{-1}F^{j+1} \tag{2.3c}$$

where h is the time step, $t^j = jh$, $x^j = x(t^j)$, $v^j = v(t^j)$, and $F^j = F(x^j)$, for any non-negative integer j. The VV integrator can also be written in a time-staggered form:

$$\begin{aligned} x^{j+1} &= x^j + h\, v^{j+1/2} \\ v^{j+3/2} &= v^{j+1/2} + h\, \mathbf{M}^{-1}F^{j+1}. \end{aligned}$$

Letting the discrete Lagrangian be the approximation of the action over a time interval of size h obtained by replacing $\dot{q}$ with a finite difference and integrating the potential with the trapezoidal rule, we have

$$L_{\mathrm{d}}(q, \tilde{q}) = h\left(\sum_{a=1}^{N} \frac{1}{2} m_a \left|\left|\frac{\tilde{q}_a - q_a}{h}\right|\right|^2 - \frac{V(q) + V(\tilde{q})}{2}\right). \tag{2.4}$$

Plugging this discrete Lagrangian (2.4) into Eq. (2.2), the momenta are given by

$$p_a^j = m_a\left(\frac{q_a^j - q_a^{j-1}}{h}\right) - \frac{h}{2}\frac{\partial V}{\partial q_a}(q_a^j) = m_a\left(\frac{q_a^{j+1} - q_a^j}{h}\right) + \frac{h}{2}\frac{\partial V}{\partial q_a}(q_a^j). \tag{2.5}$$

From Eq. (2.5) and given (q^j, p^j):

$$p_a^j = m_a\left(\frac{q_a^{j+1} - q_a^j}{h}\right) + \frac{h}{2}\frac{\partial V}{\partial q_a}(q_a^j) \;\Rightarrow\; q_a^{j+1} = q_a^j + \frac{h}{m_a}\Big(p_a^j - \frac{h}{2}\frac{\partial V}{\partial q_a}(q_a^j)\Big)$$
$$p_a^{j+1} = m_a\left(\frac{q_a^{j+1} - q_a^j}{h}\right) - \frac{h}{2}\frac{\partial V}{\partial q_a}(q_a^{j+1}).$$

Eqs. (2.3a), (2.3b) and (2.3c) are recovered provided the velocities at half steps are defined as

$$\dot{q}_a^{j+1/2} = \frac{q_a^{j+1} - q_a^j}{h},$$

the positions q and momenta p are replaced by x and mv respectively, and the equations are rewritten in vector form.

2.3 Trotter Factorization

An alternative approach to deriving the VV integrator is by using the Trotter factorization of the classical Liouville propagator. We begin by defining the Liouville operator $\mathcal{L}$ for a Hamiltonian system with N degrees of freedom in Cartesian coordinates as

$$i\mathcal{L} = \sum_{a=1}^{N}\left(\dot{q}_a\frac{\partial}{\partial q_a} + \dot{p}_a\frac{\partial}{\partial p_a}\right)$$

where q_a and p_a are the position and conjugate momentum, respectively, of the a-th degree of freedom in the system. The appearance of the imaginary number i in the definition is just a notational artifact from quantum mechanics. Now letting $\Gamma = \{q_a, p_a\}$ be the phase space vector denoting the state of the system it can be easily shown that

$$\dot{\Gamma} = i\mathcal{L}\Gamma.$$

The solution to this differential equation, given the initial state $\Gamma(0)$, can be expressed as

$$\Gamma(t) = \exp(i\mathcal{L}t)\Gamma(0) = U(t)\Gamma(0)$$

where

$$U(t) = \exp(i\mathcal{L}t)$$

is the classical Liouville propagator. However it is not obvious how to evaluate the action of $U(t)$ on the state vector $\Gamma(0)$. One approach is to decompose the Liouville operator into two parts:

$$i\mathcal{L} = i\mathcal{L}_1 + i\mathcal{L}_2$$

where

$$\begin{aligned} i\mathcal{L}_1 &= \sum_{a=1}^{N} \dot{p}_a \frac{\partial}{\partial p_a} = \sum_{a=1}^{N} F_a(q) \frac{\partial}{\partial p_a} \\ i\mathcal{L}_2 &= \sum_{a=1}^{N} \dot{q}_a \frac{\partial}{\partial q_a} = \sum_{a=1}^{N} \frac{p_a}{m_a} \frac{\partial}{\partial q_a} \end{aligned}$$

with the second equality for each operator resulting from the application of Hamilton's equations of motion. As a result for any phase space function $f(\Gamma)$ we have

$$i\mathcal{L}_1 f = \sum_{a=1}^{N} F_a(q) \frac{\partial f}{\partial p_a} \quad \text{and} \quad i\mathcal{L}_2 f = \sum_{b=1}^{N} \frac{p_b}{m_b} \frac{\partial f}{\partial q_b}.$$

An important property of this splitting is that the composition of these two operators do not commute. To see this we start by computing the two compositions

$$\begin{aligned} (i\mathcal{L}_1)(i\mathcal{L}_2) f &= (i\mathcal{L}_1) \sum_{b=1}^{N} \frac{p_b}{m_b} \frac{\partial f}{\partial q_b} \\ &= \sum_{a=1}^{N} F_a(q) \frac{\partial}{\partial p_a} \sum_{b=1}^{N} \frac{p_b}{m_b} \frac{\partial f}{\partial q_b} \\ &= \sum_{a=1}^{N} \sum_{b=1}^{N} \left[\frac{1}{m_b} F_a(q) \frac{\partial p_b}{\partial p_a} \frac{\partial f}{\partial q_b} + \frac{p_b}{m_b} F_a(q) \frac{\partial^2 f}{\partial p_a \partial q_b} \right] \end{aligned}$$

and

$$
\begin{aligned}
(i\mathcal{L}_2)\,(i\mathcal{L}_1)\,f &= (i\mathcal{L}_2)\sum_{a=1}^{N} F_a(q)\frac{\partial f}{\partial p_a} \\
&= \sum_{b=1}^{N}\frac{p_b}{m_b}\frac{\partial}{\partial q_b}\sum_{a=1}^{N} F_a(q)\frac{\partial f}{\partial p_a} \\
&= \sum_{a=1}^{N}\sum_{b=1}^{N}\left[\frac{p_b}{m_b}\frac{\partial F_a(q)}{\partial q_b}\frac{\partial f}{\partial p_a} + \frac{p_b}{m_b}F_a(q)\frac{\partial^2 f}{\partial p_a \partial q_b}\right].
\end{aligned}
$$

Letting

$$
[i\mathcal{L}_1, i\mathcal{L}_2] = (i\mathcal{L}_1)\,(i\mathcal{L}_2) - (i\mathcal{L}_2)\,(i\mathcal{L}_1)
$$

denote the commutator between the operators $i\mathcal{L}_1$ and $i\mathcal{L}_2$, it is evident that $[i\mathcal{L}_1, i\mathcal{L}_2] \neq 0$ hence the two operators do not commute. A direct consequence is that the application of this splitting to the Liouville propagator $U(t)$ leads to only a first-order accurate propagator. This can be verified by comparing the corresponding Taylor series expansions of the propagators:

$$
\begin{aligned}
U(t) &= \exp((i\mathcal{L}_1 + i\mathcal{L}_2)t) \\
&= 1 + (i\mathcal{L}_1 + i\mathcal{L}_2)t + (i\mathcal{L}_1 + i\mathcal{L}_2)^2\frac{t^2}{2} + \cdots \\
&= 1 + (i\mathcal{L}_1 + i\mathcal{L}_2)t + \frac{1}{2}\left[(i\mathcal{L}_1)^2 + (i\mathcal{L}_1)(i\mathcal{L}_2) + (i\mathcal{L}_2)(i\mathcal{L}_1) + (i\mathcal{L}_2)^2\right]t^2 + \cdots
\end{aligned}
$$

$$
\begin{aligned}
\exp(i\mathcal{L}_1 t) &= 1 + i\mathcal{L}_1 t + \frac{1}{2}(i\mathcal{L}_1)^2 t^2 + \cdots \\
\exp(i\mathcal{L}_2 t) &= 1 + i\mathcal{L}_2 t + \frac{1}{2}(i\mathcal{L}_2)^2 t^2 + \cdots \\
\exp(i\mathcal{L}_1 t)\exp(i\mathcal{L}_2 t) &= 1 + (i\mathcal{L}_1 + i\mathcal{L}_2)t + \frac{1}{2}\left[(i\mathcal{L}_1)^2 + 2(i\mathcal{L}_1)(i\mathcal{L}_2) + (i\mathcal{L}_2)^2\right]t^2 + \cdots \\
&= U(t) + [i\mathcal{L}_1, i\mathcal{L}_2]\,t^2 + \cdots \\
\exp(i\mathcal{L}_2 t)\exp(i\mathcal{L}_1 t) &= 1 + (i\mathcal{L}_1 + i\mathcal{L}_2)t + \frac{1}{2}\left[(i\mathcal{L}_1)^2 + 2(i\mathcal{L}_2)(i\mathcal{L}_1) + (i\mathcal{L}_2)^2\right]t^2 + \cdots \\
&= U(t) - [i\mathcal{L}_1, i\mathcal{L}_2]\,t^2 + \cdots
\end{aligned}
$$

Due to the non-zero commutator, both $\exp(i\mathcal{L}_1 t)\exp(i\mathcal{L}_2 t)$ and $\exp(i\mathcal{L}_2 t)\exp(i\mathcal{L}_1 t)$ exhibit an error in the second-order term.

To derive a higher-order propagator from this formalism we first observe that the leading error terms of the two decompositions have opposite signs. Hence a reasonable choice would be to take a composition of the two:

$$\begin{aligned} G(t) &= \exp(i\mathcal{L}_1(t/2))\exp(i\mathcal{L}_2(t/2))\exp(i\mathcal{L}_2(t/2))\exp(i\mathcal{L}_1(t/2)) \\ &= \exp(i\mathcal{L}_1(t/2))\exp(i\mathcal{L}_2 t)\exp(i\mathcal{L}_1(t/2)) \end{aligned}$$

where the last equality holds since $i\mathcal{L}_2$ commutes with itself. The error can once again be determined by performing a Taylor series expansion of the operators involved:

$$\begin{aligned} \exp(i\mathcal{L}_1(t/2)) &= 1 + \frac{1}{2}i\mathcal{L}_1 t + \frac{1}{8}(i\mathcal{L}_1)^2 t^2 + \cdots \\ \exp(i\mathcal{L}_2 t) &= 1 + i\mathcal{L}_2 t + \frac{1}{2}(i\mathcal{L}_2)^2 t^2 + \cdots \end{aligned}$$

Multiplying together these expansions and collecting like powers of t gives:

$$\exp(i\mathcal{L}_1(t/2))\exp(i\mathcal{L}_2 t)\exp(i\mathcal{L}_1(t/2)) = 1 + \left[\frac{1}{2}i\mathcal{L}_1 + i\mathcal{L}_2 + \frac{1}{2}i\mathcal{L}_1\right]t$$
$$+ \left[\frac{1}{8}(i\mathcal{L}_1)^2 + \frac{1}{2}(i\mathcal{L}_2)^2 + \frac{1}{8}(i\mathcal{L}_1)^2 + \frac{1}{2}(i\mathcal{L}_2)(i\mathcal{L}_1) + \frac{1}{2}(i\mathcal{L}_1)(i\mathcal{L}_2) + \frac{1}{4}(i\mathcal{L}_1)^2\right]t^2 + \cdots$$

Simplifying the coefficients we have:

$$\begin{aligned} G(t) &= 1 + (i\mathcal{L}_1 + i\mathcal{L}_2)\,t + \frac{1}{2}\left[(i\mathcal{L}_1)^2 + (i\mathcal{L}_1)(i\mathcal{L}_2) + (i\mathcal{L}_2)(i\mathcal{L}_1) + (i\mathcal{L}_2)^2\right]t^2 + \cdots \\ &= 1 + (i\mathcal{L}_1 + i\mathcal{L}_2)\,t + \frac{1}{2}\left(i\mathcal{L}_1 + i\mathcal{L}_2\right)^2 t^2 + \cdots \\ &= U(t) + O(t^3). \end{aligned}$$

Therefore $G(t)$ is a second-order accurate propagator. In addition since $G(t)$ is symmetric, it also preserves the time-reversibility of Hamiltonian dynamics, i.e. $G(-t) = G(t)$.

To construct a time integrator from the propagator $G(t)$ we appeal to the Trotter factorization [add citation]

$$U(t) = \lim_{P\to\infty} \left[G(t/P)\right]^P .$$

Taking P large then

$$U(t) \approx \left[G(t/P)\right]^P$$

which can be rewritten as follows by taking the P-th root on both sides:

$$\exp(i\mathcal{L}t/P) \approx \exp(i\mathcal{L}_1 t/2P)\exp(i\mathcal{L}_2 t/P)\exp(i\mathcal{L}_1 t/2P).$$

Interpreting the small time interval t/P as a single time step h the Trotter factorization results in the following decomposition:

$$\exp(i\mathcal{L}h) \approx \exp(i\mathcal{L}_1(h/2))\exp(i\mathcal{L}_2 h)\exp(i\mathcal{L}_1(h/2)).$$

Hence the Trotter factorization gives rise to the second-order accurate integrator

$$G(h) = \exp(i\mathcal{L}_1(h/2))\exp(i\mathcal{L}_2 h)\exp(i\mathcal{L}_1(h/2))$$

where $G(h)$ is a discrete propagator that advances the system forward by the time step h.

To obtain the VV integrator from the Trotter factorization recall that

$$i\mathcal{L}_1 = \sum_{a=1}^{N} F_a(q)\frac{\partial}{\partial p_a}, \quad i\mathcal{L}_2 = \sum_{a=1}^{N} \frac{p_a}{m_a}\frac{\partial}{\partial q_a}.$$

Then the corresponding discrete propagator is

$$G(h) = \exp\left[\frac{h}{2}\sum_{a=1}^{N} F_a(q)\frac{\partial}{\partial p_a}\right]\exp\left[h\sum_{a=1}^{N}\frac{p_a}{m_a}\frac{\partial}{\partial q_a}\right]\exp\left[\frac{h}{2}\sum_{a=1}^{N} F_a(q)\frac{\partial}{\partial p_a}\right].$$

To determine the action of each of these three operators on the state vector Γ we begin by observing that

$$\left[\exp\left(c\frac{\partial}{\partial x}\right)\right]f(x) = f(x+c)$$

where c is independent of x. This result can be shown by expressing the left-hand side as a power series expansion

$$\begin{aligned}\left[\exp\left(c\frac{\partial}{\partial x}\right)\right]f(x) &= \left[1+\left(c\frac{\partial}{\partial x}\right)+\frac{1}{2}\left(c\frac{\partial}{\partial x}\right)^2+\cdots\right]f(x)\\ &= f(x)+c\frac{\partial}{\partial x}f(x)+\frac{1}{2}c^2\frac{\partial^2}{\partial x^2}f(x)+\cdots\\ &= \sum_{i=0}^{\infty}\frac{f^{(i)}(x)}{i!}c^i\end{aligned}$$

and recognizing that the final expression is simply the Taylor series expansion of $f(x+c)$ about x. If $f(x)$ is replaced by $g(y)$ where y is independent of x then

$$\left[\exp\left(c\frac{\partial}{\partial x}\right)\right]g(y) = \left[1+\left(c\frac{\partial}{\partial x}\right)+\frac{1}{2}\left(c\frac{\partial}{\partial x}\right)^2+\cdots\right]g(y) = g(y).$$

The application of $G(h)$ to the state at time jh $\Gamma(jh) = \{q_a^j, p_a^j\}$ can be broken down into three steps. First:

$$\exp\left[\frac{h}{2}\sum_{a=1}^{N}F_a(q)\frac{\partial}{\partial p_a}\right]\{q_a^j, p_a^j\} = \{q_a^j, p_a^j + \frac{h}{2}F_a(q^j)\}.$$

Letting

$$p_a^{j+1/2} = p_a^j + \frac{h}{2}F_a(q^j)$$

the second step can then be expressed as:

$$\exp\left[h\sum_{a=1}^{N}\frac{p_a}{m_a}\frac{\partial}{\partial q_a}\right]\{q_a^j, p_a^{j+1/2}\} = \{q_a^j + \frac{h}{m_a}p_a^{j+1/2}, p_a^{j+1/2}\}.$$

Now defining

$$q_a^{j+1} = q_a^j + \frac{h}{m_a} p_a^{j+1/2}$$

the final step, similar to the first, gives:

$$\exp\left[\frac{h}{2}\sum_{a=1}^{N} F_a(q)\frac{\partial}{\partial p_a}\right]\{q_a^{j+1}, p_a^{j+1/2}\} = \{q_a^{j+1}, p_a^{j+1/2} + \frac{h}{2}F_a(q^{j+1})\}.$$

In vector form the integrator can be summarized as

$$\begin{aligned} p^{j+1/2} &= p^j + \frac{h}{2}F^j \\ q^{j+1} &= q^j + h\,\mathbf{M}^{-1}p^{j+1/2} \\ p^{j+1} &= p^{j+1/2} + \frac{h}{2}F^{j+1}. \end{aligned}$$

Replacing the position vector q with x and the momentum vector p with $\mathbf{M}v$, the VV integrator given by Eq. (2.3) is recovered.

2.4 Derivation of r-RESPA

In this section we will describe how the multiple time stepping method r-RESPA [90, 38] can be derived from the Trotter factorization, as shown in [90]. The Trotter factorization can be extended to systems whose potential $V(q)$ can be written as a sum of a fast-varying potential $V^{\text{fast}}(q)$ and a slow-varying potential $V^{\text{slow}}(q)$ as follows. Since $V^{\text{slow}}(q)$ is a smooth potential, we would like to integrate it less frequently by choosing a larger time step than that for $V^{\text{fast}}(q)$. To generate a multiple time stepping method we start with the following decomposition of the Liouville operator:

$$i\mathcal{L} = i\mathcal{L}^{\text{slow}} + i\mathcal{L}^{\text{fast}}$$

where

$$\begin{aligned} i\mathcal{L}^{\text{slow}} &= \sum_{a=1}^{N} F_a^{\text{slow}}(q)\frac{\partial}{\partial p_a} \\ i\mathcal{L}^{\text{fast}} &= \sum_{a=1}^{N} F_a^{\text{fast}}(q)\frac{\partial}{\partial p_a} + \sum_{a=1}^{N} \frac{p_a}{m_a}\frac{\partial}{\partial q_a} \end{aligned}$$

Noting that the operator $i\mathcal{L}^{\text{fast}}$ is the same as that for a single-potential system, the Trotter factorization can be applied to this propagator to give

$$G^{\text{fast}}(h) = \exp\left[\frac{h}{2}\sum_{a=1}^{N} F_a^{\text{fast}}(q)\frac{\partial}{\partial p_a}\right] \exp\left[h\sum_{a=1}^{N}\frac{p_a}{m_a}\frac{\partial}{\partial q_a}\right] \exp\left[\frac{h}{2}\sum_{a=1}^{N} F_a^{\text{fast}}(q)\frac{\partial}{\partial p_a}\right]$$

where h is the time step for $V^{\text{fast}}(q)$. Now observing that $i\mathcal{L}$ is itself a decomposition of two operators, the Trotter factorization can be used again to obtain a propagator for the entire two-potential system:

$$G(\Delta t) = \exp\left[\frac{\Delta t}{2}\sum_{a=1}^{N} F_a^{\text{slow}}(q)\frac{\partial}{\partial p_a}\right] \exp\left(i\mathcal{L}^{\text{fast}}\Delta t\right) \exp\left[\frac{\Delta t}{2}\sum_{a=1}^{N} F_a^{\text{slow}}(q)\frac{\partial}{\partial p_a}\right]$$

with Δt corresponding to the time step for $V^{\text{slow}}(q)$. If Δt is chosen to be an integer multiple of h, say $\Delta t = Mh$, then we have

$$\begin{aligned} G(\Delta t) &= \exp\left[\frac{\Delta t}{2}\sum_{a=1}^{N} F_a^{\text{slow}}(q)\frac{\partial}{\partial p_a}\right] \exp\left(i\mathcal{L}^{\text{fast}}Mh\right) \exp\left[\frac{\Delta t}{2}\sum_{a=1}^{N} F_a^{\text{slow}}(q)\frac{\partial}{\partial p_a}\right] \\ &= \exp\left[\frac{\Delta t}{2}\sum_{a=1}^{N} F_a^{\text{slow}}(q)\frac{\partial}{\partial p_a}\right] \left[\exp\left(i\mathcal{L}^{\text{fast}}h\right)\right]^M \exp\left[\frac{\Delta t}{2}\sum_{a=1}^{N} F_a^{\text{slow}}(q)\frac{\partial}{\partial p_a}\right] \\ &= \exp\left[\frac{\Delta t}{2}\sum_{a=1}^{N} F_a^{\text{slow}}(q)\frac{\partial}{\partial p_a}\right] \left[G^{\text{fast}}(h)\right]^M \exp\left[\frac{\Delta t}{2}\sum_{a=1}^{N} F_a^{\text{slow}}(q)\frac{\partial}{\partial p_a}\right]. \end{aligned}$$

The integrator that results from this sequence of propagators is exactly r-RESPA for a system with two potentials [90]. By determining the action of each exponential in the factorization, Algorithm 1 is obtained.

Algorithm 1 r-RESPA for Two Potentials

```
Input: $x^0$; $v^0$; time steps $h$ and $\Delta t = Mh$
Output: $(x^{\{i\}}, v^{\{i\}})$, for all times $t^i = ih$

**Initialization**
$i = 0$
$x = x^0$; $v = v^0$
Compute forces $F^{\text{slow}}(x)$ and $F^{\text{fast}}(x)$

**Integrate the system over the time interval** $[0, T]$ **where** $T = P\Delta t$
**for all** $p = 1$ to $P$ **do** {Loop over slow time step $\Delta t$}
    **for all** $a$ **do** {Half-kick for slow potential}
        $v_a = v_a + \frac{\Delta t}{2} \frac{F_a^{\text{slow}}}{m_a}$
    **end for**
    **for all** $m = 1$ to $M$ **do** {Loop over fast time step $h$}
        **for all** $a$ **do** {Half-kick for fast potential}
            $v_a = v_a + \frac{h}{2} \frac{F_a^{\text{fast}}}{m_a}$
        **end for**
        $x = x + h\,v$ {Drift}
        Compute $F^{\text{fast}}(x)$
        **for all** $a$ **do** {Half-kick for fast potential}
            $v_a = v_a + \frac{h}{2} \frac{F_a^{\text{fast}}}{m_a}$
        **end for**
        $i = i + 1$;
        $x^i = x$; $v^i = v$
    **end for**
    Compute $F^{\text{slow}}(x)$
    **for all** $a$ **do** {Half-kick for slow potential}
        $v_a = v_a + \frac{\Delta t}{2} \frac{F_a^{\text{slow}}}{m_a}$
    **end for**
    $v^i = v$
**end for**
```

The operator splitting presented in this section can be applied recursively to construct a multiple time step integrator for systems in which the potential $V(q)$ can written as a sum of K potentials

$$V(q) = \sum_{k=1}^{K} V_k(q).$$

Here we assume that the potentials $V_k(q)$ become smoother as k increases hence larger time steps can be chosen for larger k. By choosing all successive time step pairs to be in integer relation, this results in r-RESPA for a K-potential system. In the following section we will show that r-RESPA can be recovered from the variational framework by observing that it is a specific instance of AVI, namely when the time steps are chosen to be in integer ratios.

2.5 Derivation of AVI

We discuss next the derivation of AVI. The types of asynchronous discretizations discussed herein are applicable to situations in which the potential $V(q)$ can be written as the sum of K potentials:

$$V(q) = \sum_{k=1}^{K} V_k(q).$$

In the context of finite-element discretizations of continuum mechanics equations this decomposition is naturally accomplished on an element-by-element basis, while in the context of molecular dynamics for large proteins it is often achieved by splitting the forces into strong, short-ranged ones and weak, long-ranged ones. In these situations it is possible to integrate each one of the potentials V_k with a different time step, obtaining in this way a more efficient algorithm for a given desired accuracy.

The idea is then to assign to each potential V_k a sequence of times $\{0 = t_k^0 < \ldots < t_k^{M_k} = T\}$. Additionally, we construct the sequence of all times in the system $\{\theta^0 < \theta^1 < \ldots < \theta^M\}$ by lumping together all potential times in a strictly increasing sequence; see the example in Fig. 2.1. As before, the position of the system at time θ^i

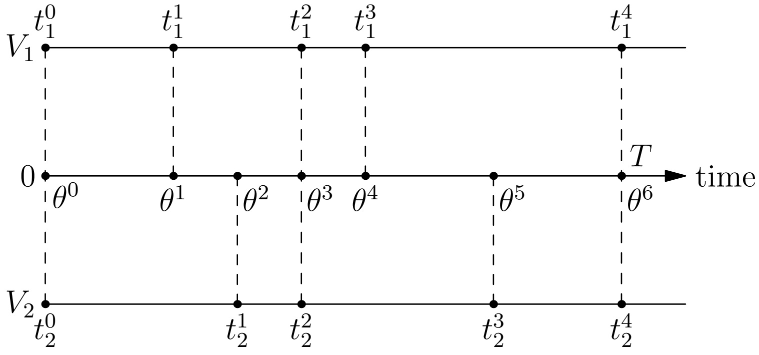


Figure 2.1: Example of a time discretization for AVI. In this case a split into two potential energy functions is adopted. The times for potential V_1 are $\{0, t_1^1, t_1^2, t_1^3, t_1^4\}$ and those of potential V_2 are $\{0, t_2^1, t_2^2, t_2^3, t_2^4\}$. The resulting set of all time steps in the system is $\{\theta^0 = t_1^0 = t_2^0,\ \theta^1 = t_1^1,\ \theta^2 = t_2^1,\ \theta^3 = t_1^2 = t_2^2,\ \theta^4 = t_1^3,\ \theta^5 = t_2^3,\ \theta^6 = t_1^4 = t_2^4 = T\}$.

is denoted by q^i, and a discrete trajectory is the sequence of positions $\{q^0, \ldots, q^M\}$. For each time θ^i we define the set $\mathcal{K}(i)$ as:

$$\mathcal{K}(i) = \{k \mid \exists j,\ t_k^j = \theta^i\}.$$

For each $k \in \mathcal{K}(i)$, we can define:

$$h_k^{i+1/2} \overset{\text{def}}{=} t_k^{j+1} - t_k^j \quad \text{and} \quad h_k^{i-1/2} \overset{\text{def}}{=} t_k^j - t_k^{j-1},$$

where $t_k^j = \theta^i$.

The discrete Lagrangian for AVI is:

$$L_{\mathrm{d}}(q, \tilde{q}, i) = \sum_{a=1}^{N} \frac{1}{2} m_a \Delta\theta \left\| \frac{\tilde{q}_a - q_a}{\Delta\theta} \right\|^2 - \sum_{k\in\mathcal{K}(i)} \frac{h_k^{i+1/2}}{2} V_k(q) - \sum_{k\in\mathcal{K}(i+1)} \frac{h_k^{i+1/2}}{2} V_k(\tilde{q}), \quad (2.6)$$

with $\Delta\theta = \theta^{i+1} - \theta^i$. The discrete action sum follows as

$$S_{\mathrm{d}} = \sum_{i=0}^{M-1} L_{\mathrm{d}}(q^i, q^{i+1}, i).$$

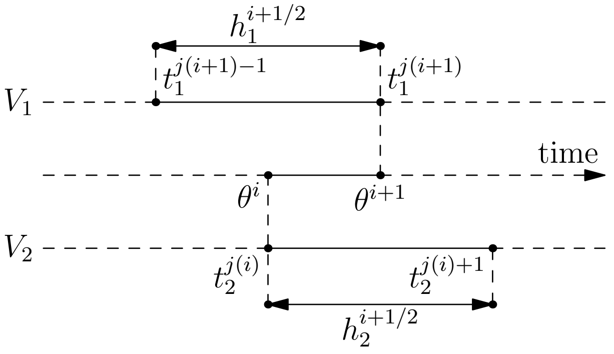


Figure 2.2: Graphical interpretation of the arguments of the discrete Lagrangian for AVI, Eq. (2.6), for a generic time interval (θ^i, θ^{i+1}). The first term on the right-hand-side of Eq. (2.6) approximates the action stemming from the kinetic energy during (θ^i, θ^{i+1}). In contrast, each one of the two remaining terms account for one half of the contribution of potentials at time θ^i and θ^{i+1}.

A graphical interpretation of this discrete Lagrangian is shown in Fig. 2.2.

One of the noteworthy features of this presentation, in contrast to that in [58], is that the discrete Lagrangian is not a consistent approximation of the action during a time interval (θ^i, θ^{i+1}), in the sense explained in [66]. Nonetheless, S_d is still a consistent approximation of the action over the whole trajectory during the time interval $[0, T]$. This is evident from rearranging the terms in the sum, namely,

$$S_\mathrm{d} = \sum_{a=1}^{N} \sum_{i=0}^{M-1} \frac{1}{2} m_a (\theta^{i+1} - \theta^i) \left|\left| \frac{q_a^{i+1} - q_a^i}{\theta^{i+1} - \theta^i} \right|\right|^2 - \sum_{k=1}^{K} \sum_{j=0}^{M_k - 1} (t_k^{j+1} - t_k^j) \frac{V_k(q^{k,j+1}) + V_k(q^{k,j})}{2},$$

where $q^{k,j}$ is the position at time t_k^j. The discrete Euler-Lagrange equations take the form:

$$m_a \dot{q}_a^{i+1/2} - m_a \dot{q}_a^{i-1/2} = - \sum_{k \in \mathcal{K}(i)} \frac{h_k^{i-1/2} + h_k^{i+1/2}}{2} \frac{\partial V_k}{\partial q_a}(q^i) \tag{2.7}$$

where

$$\dot{q}_a^{i+1/2} \overset{\mathrm{def}}{=} \frac{q_a^{i+1} - q_a^i}{\theta^{i+1} - \theta^i}.$$

Eq. (2.2) defines the momenta $\{p^0, \dots, p^M\}$, to wit

$$p_a^i = m_a \dot{q}_a^{i-1/2} - \sum_{k \in \mathcal{K}(i)} \frac{h_k^{i-1/2}}{2} \frac{\partial V_k}{\partial q_a}(q^i) = m_a \dot{q}_a^{i+1/2} + \sum_{k \in \mathcal{K}(i)} \frac{h_k^{i+1/2}}{2} \frac{\partial V_k}{\partial q_a}(q^i).$$

This derivation of AVI and the method itself differ slightly from those provided in [58, 59]. The first difference with [58, 59] is the precise definition of the map $(q^i, p^i) \mapsto (q^{i+1}, p^{i+1})$, which was absent in the previous references. One key consequence is that it makes the symplectic nature of the asynchronous discretization evident: due to the two-point discrete Lagrangian in Eq. (2.6) and its associated definition of the momenta, the resulting map is symplectic. In contrast, in [58], it is shown that AVI is a multi-symplectic algorithm, which is a natural concept in the context of continuum mechanics, but that lacks a clear or traditional interpretation in the ordinary differential equations setting.

The second difference is the use of a trapezoidal rule to approximate the action integral within each elemental time step, as opposed to the rectangle rule adopted in [58, 59]. A consequence of this choice is the appearance of the average of two consecutive time step sizes in the discrete Euler-Lagrange equations (2.7). This difference vanishes when the time step for each potential is constant.

Finally, by reverting to the x and v notation adopted at the beginning of the section AVI reads

$$v_a^{i+1/2} = v_a^i - \frac{1}{m_a} \sum_{k \in \mathcal{K}(i)} \frac{h_k^{i+1/2}}{2} \frac{\partial V_k}{\partial x_a}(x^i), \tag{2.8a}$$

$$x^{i+1} = x^i + (\theta^{i+1} - \theta^i)\, v^{i+1/2} \tag{2.8b}$$

$$v_a^{i+1} = v_a^{i+1/2} - \frac{1}{m_a} \sum_{k \in \mathcal{K}(i+1)} \frac{h_k^{i-1/2}}{2} \frac{\partial V_k}{\partial x_a}(x^{i+1}), \tag{2.8c}$$

which reduces to VV when all time steps are identical.

It can also be verified that AVI is a generalization of r-RESPA. To this end, it is enough to choose the time steps for each potential such that $h_{k+1}/h_k = r_k$ is an integer, for all $k \geq 1$. In that case, Eq. (2.8a)–(2.8c) can be implemented as shown in Algorithm 3 in Appendix A.1. This algorithm is identical to r-RESPA.

2.6 Algorithm Implementation of AVI

Since each of the potentials V_k has a different time step, a priority queue is used to determine the order in which the potentials are evaluated. The elements of this priority queue have the form (t_k^j, k), where t_k^j is the next time at which potential V_k needs to be evaluated, with the elements sorted in ascending order with respect to t_k^j. In case of equality of t_k^j for different ks, the ordering does not matter. As a result the element at the top gives the time of the next potential evaluation and the indices j and k corresponding to the time t_k^j. The AVI routine in Algorithm 2 below is a possible implementation, best tailored for problems with only a few different potentials with essentially all degrees of freedom as the arguments for each one of them. This is a typical situation encountered in the simulation of macromolecules with molecular dynamics. In contrast, a version of the algorithm better suited for finite-element-like simulations has already been introduced in [58]. In this latter case there are a large number of different potentials with only a few arguments each.

Algorithm 2 AVI Algorithm

```
Input: $\theta^0$; $x^0$; $v^0$; set of all potential times $t_{\{k\}}^{\{j\}}$
Output: $(\theta^{\{i\}}, x^{\{i\}}, v^{\{i\}})$, for all $i$

**Initialization**
$i = 0$
$v = v^0$; $x = x^0$; $\theta^{\text{old}} = \theta^0$
$F^{1/2} = F^{-1/2} = 0$
**for all** $k$ **do**
  Push $(t_k^0, k)$ into the priority queue $Q$
  $M_k =$ size of array $t_k^{\{j\}}$
**end for**

**Integrate the system over the time interval** $[0, T]$
**while** priority queue is not empty **do**
  Pop the top element $(t_k^j, k)$ from $Q$
  $\theta^{\text{new}} = t_k^j$
  **if** $\theta^{\text{new}} > \theta^{\text{old}}$ **then**
    **for all** $a$ **do** {Half-kick}
      $v_a = v_a + \frac{1}{2}\frac{F_a^{-1/2}}{m_a}$
    **end for**
    $x^i = x$; $v^i = v$; $\theta^i = \theta^{\text{old}}$
    $i = i + 1$
    **for all** $a$ **do** {Half-kick}
      $v_a = v_a + \frac{1}{2}\frac{F_a^{1/2}}{m_a}$
    **end for**
    $x = x + (\theta^{\text{new}} - \theta^{\text{old}})\, v$ {Drift}
    $F^{1/2} = F^{-1/2} = 0$
  **end if**
  $\theta^{\text{old}} = \theta^{\text{new}}$
  **if** $j > 0$ **then**
    $F^{-1/2} = F^{-1/2} - (t_k^j - t_k^{j-1})\nabla V_k(x)$
  **end if**
  **if** $j < M_k$ **then**
    $F^{1/2} = F^{1/2} - (t_k^{j+1} - t_k^j)\nabla V_k(x)$
    Push $(t_k^{j+1}, k)$ into the priority queue $Q$
  **end if**
**end while**
**for all** $a$ **do** {Half-kick}
  $v_a = v_a + \frac{1}{2}\frac{F_a^{-1/2}}{m_a}$
**end for**
$x^i = x$; $v^i = v$; $\theta^i = \theta^{\text{old}}$
```

Chapter 3

Stability of Variational Integrators

The adoption of multiple time step integrators can provide substantial computational savings for mechanical systems with multiple time scales. However, the scope of these savings may be limited by the range of allowable time step choices. In this chapter we analyze the linear stability of variational integrators. We perform a detailed analysis for the case of a one-dimensional particle moving under the action of a soft and a stiff quadratic potential, integrated with two time steps in rational ratios. In this case, we provide sufficient conditions for the stability of the method. These generalize to the fully asynchronous AVI case the results obtained for synchronous multiple time stepping schemes, such as r-RESPA, which show resonances when the larger time step is a multiple of the effective half-period of the stiff potential. Additionally, we numerically investigate the appearance of instabilities. Based on the experimental observations, we conjecture the existence of a dense set of unstable time steps when arbitrary rational ratios of time steps are considered. In this way, unstable schemes for arbitrarily small time steps can be obtained. However, the vast majority of these instabilities are extremely weak and do not present an obstacle to the use of these integrators. We then applied these results to analyze the stability of multiple time step integrators in the more complex mechanical systems arising in molecular dynamics and solid dynamics. We explained why strong resonances are ubiquitously found in the former, while rarely encountered in the latter.

3.1 Introduction

It has been long recognized that r-RESPA, whose formulation was described in Sec. 2.4, can display resonance instabilities, especially in the context of molecular dynamics simulations [50, 3, 79, 81, 63, 9]. These resonances severely limit the relative size of the time steps. Several approaches have been proposed to reduce these instabilities. Schlick et al. [3, 4, 77] used Langevin dynamics along with extrapolative methods. An appropriate choice of the friction coefficient in the Langevin equation stabilizes the trajectories even with very large time steps. The isokinetic Nosé-Hoover chain RESPA proposed by Minary et al. [69] is another method that produces stable trajectories for large time steps. Mollified versions of r-RESPA such as MOLLY were developed by Izaguirre et al. [51]. MOLLY retains instabilities but they appear for larger values of the time steps (up to 50% greater). When the fast potentials are assumed to be quadratic, an elegant procedure, called the two-force method [40, 41], removes all resonances and captures the correct coupling between slow and fast forces (see also [43]). However, the extension of this scheme to situations with more than two time steps has not been reported so far.

Contrary to molecular dynamics, resonance instabilities have not hampered the use of multiple time step methods in computational solid mechanics. Stability analyses for several subcycling methods have been performed [46, 6, 21, 20, 19]. In particular, many interesting aspects of the stability of some subcycling methods have been highlighted in [20, 21]. Therein, it was posited that the reason behind the successful performance of multiple time step methods in solid dynamics is that the set of resonant time steps is very small. Consequently (within some reasonable stability considerations), it is unlikely in practice that resonant time steps will be chosen. Even though the algorithms analyzed therein are not symplectic, and hence essentially differ from AVI, we shall see that these same observations are valid for AVI.

The key distinctive feature of AVI over r-RESPA or many of the subcycling algorithms is that time steps in arbitrary ratios can be considered. This extra degree of freedom becomes very useful in FE simulations, since time steps can be made to vary smoothly throughout the mesh. In the molecular dynamics context, this freedom enables the adoption of more general decompositions of the potential energy, each one with a characteristic time and length scale. In fact, as we shall discuss in subsequent sections, AVI generalizes r-RESPA to arbitrary (instead of integer) time step ratios.

The complex stability considerations found in previous multiple time step methods with integer time step ratios is enriched when rational time step ratios are considered. The description of these novel features and their analysis are one of the two main contributions in this chapter. The second key contribution is to utilize this analysis and some carefully crafted numerical experiments to explain the dichotomy in the behavior of AVI between molecular and solid dynamics simulations.

The key contributions found in this chapter are:

1. A linear stability analysis of AVI when two time steps h_1 and h_2 are used to integrate a one-degree of freedom harmonic oscillator whose potential energy has been split into a stiff and a soft part. We provide, in the form of Proposition 1, a bound on the trace of the amplification matrix for integrators in which h_2/h_1 is a rational number. As a corollary, a sufficient condition for the stability of the integrator follows. The resulting possible unstable time step combinations generalize those obtained for r-RESPA in [3, 77] for the same system.

2. A conjecture that the set of unstable time steps is dense and that arbitrary small unstable time steps exist. This conjecture is suggested by the theoretical analysis and numerical experiments. Systematic numerical tests in which all unstable time steps were obtained along lines of the form $h_2 = (p/q)h_1$ (p and q integers) strongly support this conjecture. Most of these resonances, however, are extremely weak and would require millions of time steps or more to be observed, so they have no practical implications.

3. A numerical study of the location of the strongest instabilities as a function of h_1 and h_2, again for a harmonic oscillator, which is similar to that presented in [32]. We propose a criterion to characterize the location of the strongest resonances, and verify its validity by predicting the location of the most important resonances in the h_1-h_2 plane.

4. We demonstrate, through numerical examples and some analysis, that the weak long-range forces often present in molecular dynamics are the key culprit for the stringent stability limitations of AVI. In the context of solid dynamics, the local coupling between elements leads to a weak coupling between stiff and soft regions when these vary smoothly in space. We show that as a consequence the set of time steps leading to unstable schemes is very small, explaining why these resonance instabilities that are pervasive in molecular dynamics are only seldom observed in FE simulations with AVI.

Perhaps surprisingly, most of the features in the molecular dynamics and solid mechanics examples can be simply understood with the analysis of the one-degree of freedom system. The integration of weak long-range forces with a large time step near an integer multiple of the half-period of one of the natural modes in the system leads to a resonance instability. This is manifested as an exponential growth of the amplitude of that mode. The width of the resonance for each mode, i.e. the range of time steps for which a resonance is encountered, decreases with the stiffness of the long-range forces. Consequently, the weaker the forces the more difficult it is to excite a resonant mode, and the slower the exponential growth is. In molecular dynamics, the fact that long-range forces interact with every single degree of freedom in the molecule leads to wide resonance intervals. Since in large molecular systems the set of natural frequencies is dense, it is almost certain that beyond a certain threshold one of these frequencies will satisfy the resonance condition. The same limitation is found in most other methods such as r-RESPA.

In contrast, in solid dynamics, soft elements integrated with large time steps also have the possibility of inducing a resonance in one or more of the high-frequency or stiff natural modes of the discrete structure. However, numerical examples and analytical considerations show that the amplitude of the stiff modes decays exponentially fast in a soft region; this leads to a very weak coupling between stiff and soft elements and to very narrow resonance intervals. These resonances are so difficult to encounter even when an explicit effort to find them is made, that they effectively have no practical implications, leading to a robust multiple time step integration algorithm. For the same reasons, it follows that it is safer to pick time steps which vary smoothly in space, so that sudden transitions from stiff to soft materials do not induce resonant instabilities. This is consistent with the findings in [21] for other multiple time step integrators in solid dynamics.

In Section 3.2, the stability of r-RESPA discretizations for a harmonic oscillator is reviewed [3, 77], since it is essential for the subsequent analysis of the more complex AVI case in Section 3.3. Therein, the analysis proceeds by studying the stability behavior of AVI in the case of a harmonic oscillator formed by two springs, in which one is very soft relative to the other. The analysis reveals a sufficient condition for stability. We are not able to specify the behavior of the discretization when these conditions are not satisfied, but numerical experiments show that instabilities are systematically encountered in at least part of each connected time step interval in which these conditions are not met. A study of our conjecture that the set of unstable time steps is dense follows. In particular, we provide specific examples of extremely small time steps which lead to an exponential growth of the energy. The study of the location of the strongest resonances for the harmonic oscillator is also presented here. Section 3.4 contains the study of resonance instabilities for AVI in molecular dynamics and solid dynamics simulations. A summary is given in Section 3.5.

3.2 Stability of multi-step integrators

We begin by analyzing the stability of VV and r-RESPA. Similar analyses have been published elsewhere [77, 3]. However since we will show that the results for AVI extend this analysis, we briefly recall the main results regarding VV and r-RESPA (see e.g. [77, 3] for a similar analysis).

Consider a system of n first-order ODEs:

$$\dot{x} = \mathbf{A}x, \quad x(0) = x_0.$$

Let $\mathbf{Q}$ be the propagation matrix representing the numerical integrator $x^{j+1} = \mathbf{Q}\, x^j$, where $\{x^0, x^1, \ldots, x^M\}$ is a time discretization of $x(t)$. Then the integrator represented by $\mathbf{Q}$ is stable if and only if its eigenvalues $\lambda_i(\mathbf{A})$ satisfy

$$|\lambda_i| \leq 1 \tag{3.1}$$

and are semi-simple[1] when equal.

We should note that a linear stability analysis does not necessarily capture all possible instabilities. Non-linearities can play an important role in rendering linearly stable schemes unstable, as shown in [80].

We now focus on the propagation matrix $\mathbf{Q}_{\mathrm{VV}}$ for a 1-D harmonic oscillator

$$\ddot{x} + \Lambda x = 0, \quad x(0) = x_0, \; \dot{x}(0) = v_0$$

integrated with VV. The matrix $\mathbf{Q}_{\mathrm{VV}}$ acts on phase-space variables x and $\dot{x}$. It is equal to:

$$\mathbf{Q}_{\mathrm{VV}} = \begin{bmatrix} 1 - \frac{h^2}{2}\Lambda & h \\ -h\Lambda\left(1 - \frac{h^2}{4}\Lambda\right) & 1 - \frac{h^2}{2}\Lambda \end{bmatrix}.$$

It can be shown that the eigenvalues of $\mathbf{Q}_{\mathrm{VV}}$ satisfy the stability condition if and only if

$$h < \frac{2}{\sqrt{\Lambda}}. \tag{3.2}$$

The range of stable time steps can also be found by looking at the shadow Hamiltonian for the numerical integrator. Obtained from backward error analysis, the shadow Hamiltonian for a symplectic integrator is such that the trajectory it generates matches exactly the numerical integrator at each time step. Shadow Hamiltonians are presented and discussed in greater detail in [41, 56, 68, 42]. Here the shadow Hamiltonian will be constructed for $\mathbf{Q}_{\mathrm{VV}}^2$ instead of $\mathbf{Q}_{\mathrm{VV}}$. The reason is that $\mathbf{Q}_{\mathrm{VV}}$ may have negative eigenvalues in which case the shadow Hamiltonian is complex-valued. However by considering $\mathbf{Q}_{\mathrm{VV}}^2$ the eigenvalues are always positive and the resulting shadow Hamiltonian is always real. The shadow Hamiltonian for the integrator with

[1] An eigenvalue is semi-simple if the number of independent eigenvectors corresponding to that eigenvalue is equal to its algebraic multiplicity.

propagator matrix $\mathbf{Q}_{\mathrm{VV}}^2$ is

$$\tilde{H}_{\mathrm{VV}}(q,p_q) = \begin{cases} \left(\frac{p_q^2}{2\gamma} + \frac{1}{2}\gamma\Lambda q^2\right)\frac{\cos^{-1}\left(1-\frac{h^2}{2}\Lambda\right)}{h\sqrt{\Lambda}} & \text{if } h < 2/\sqrt{\Lambda} \\ -\frac{1}{2}p_q^2 & \text{if } h = 2/\sqrt{\Lambda} \\ \left(-\frac{p_q^2}{2\gamma} + \frac{1}{2}\gamma\Lambda q^2\right)\frac{\cosh^{-1}\left(\frac{h^2}{2}\Lambda-1\right)}{h\sqrt{\Lambda}} & \text{if } h > 2/\sqrt{\Lambda} \end{cases}$$

where p_q is the conjugate momentum of the spatial coordinate q and

$$\gamma = \sqrt{\left|1 - \left(\frac{h}{2}\sqrt{\Lambda}\right)^2\right|}.$$

When $h < 2/\sqrt{\Lambda}$ the shadow Hamiltonian agrees with the result from [68]. Noticing that the level sets of $\tilde{H}_{\mathrm{VV}}$ are ellipses if $h < 2/\sqrt{\Lambda}$ we conclude that VV is stable in this regime. However if $h = 2/\sqrt{\Lambda}$ the level sets of $\tilde{H}_{\mathrm{VV}}$ are now lines whereas for $h > 2/\sqrt{\Lambda}$ the level sets are hyperbolas. In both cases these contours correspond to unstable trajectories. Therefore VV is unstable if $h \geq 2/\sqrt{\Lambda}$.

To study the stability of multiple time step integrators, we start by examining the basic resonance mechanism for these integrators. Consider a 1-D harmonic oscillator with the splitting $\Lambda = \Lambda_1 + \Lambda_2$ where $\Lambda_1 \geq \Lambda_2 > 0$. Hereafter the spring with spring constant Λ_1 will be referred to as the stiff spring and Λ_2 as the soft spring. We consider then the case in which the stiff spring is integrated exactly. This results in a sinusoidal trajectory in time with constant energy as long as the soft spring is not accounted for. The time-integration scheme for the soft spring modifies this trajectory by imparting an impulse (or "kick") on the oscillator at time intervals of length h_2, which makes the momentum instantaneously jump to a new value. Between any two consecutive impulses, the trajectory is still sinusoidal in time with constant energy. If h_2 happens to be equal to an integer multiple of the half-period of the fast oscillator, then a resonance occurs, as clearly illustrated by the phase diagram in Fig. 3.1. In this case the initial conditions are $x = 1$ and $\dot{x} = 0$. The soft spring impulse is then always applied when $x = 1$, $\dot{x} \leq 0$ or $x = -1$, $\dot{x} \geq 0$. In both cases, the sign of the force is such that it results in a net growth in speed, bringing the oscillator to continue moving on a larger ellipse with increased energy. This leads to a resonant behavior. An analog behavior will be observed for values of h_2 that are close to but not exactly equal to half of the period of the stiff oscillator, as we shall see next.

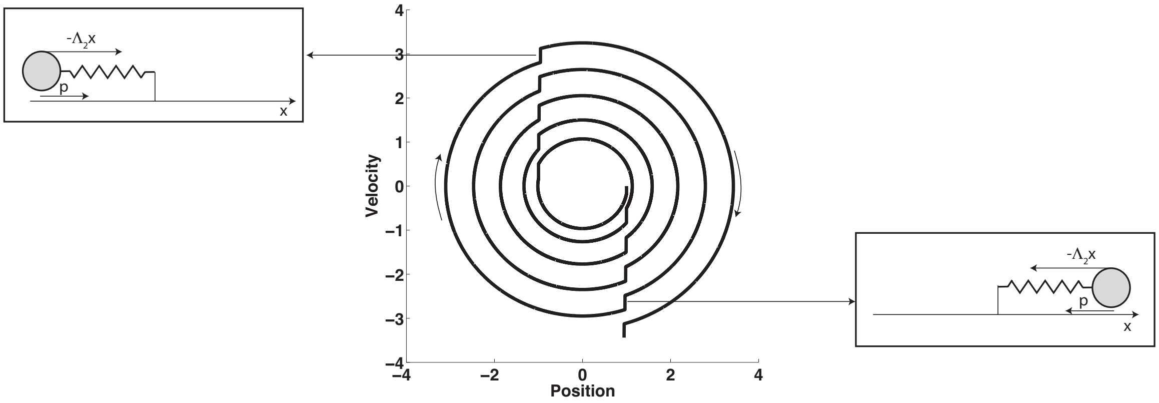


Figure 3.1: Phase plane diagram of harmonic oscillator hit with a velocity impulse every half-period. The initial conditions are $x = 1$ and $\dot{x} = 0$. Notice that in this case the impulses result in a net energy growth, as evidenced by the radius of the circle representing the trajectory of the harmonic oscillator.

With this resonance mechanism in mind we now proceed to examine the effect of integrating the stiff spring with a discrete time step h_1. Consider the case in which $h_2 = ph_1$ where p is an integer. A single integration step of length h_2 can be decomposed as (see Eqs. (2.8)):

$$\begin{bmatrix} x^{j+1} \\ v^{j+1} \end{bmatrix} = \mathbf{V}_{\text{slow}} \left(\mathbf{Q}_{\text{fast}}\right)^p \mathbf{V}_{\text{slow}} \begin{bmatrix} x^j \\ v^j \end{bmatrix} \overset{\text{def}}{=} \mathbf{Q}_{\text{r-RESPA}} \begin{bmatrix} x^j \\ v^j \end{bmatrix}$$

where

$$\mathbf{V}_{\text{slow}} = \begin{bmatrix} 1 & 0 \\ -\frac{h_2}{2}\Lambda_2 & 1 \end{bmatrix}$$

and

$$\mathbf{Q}_{\text{fast}} = \begin{bmatrix} 1 - \frac{h_1^2}{2}\Lambda_1 & h_1 \\ -h_1\Lambda_1\left(1 - \frac{h_1^2}{4}\Lambda_1\right) & 1 - \frac{h_1^2}{2}\Lambda_1 \end{bmatrix}.$$

Since $\mathbf{Q}_{\text{r-RESPA}}$ is the product of matrices each with determinant 1, its determinant is also 1. Therefore when $|\mathrm{Tr}(\mathbf{Q}_{\text{r-RESPA}})| < 2$, the two eigenvalues are distinct complex conjugates lying on the unit circle. Hence the integrator is stable.

When $|\mathrm{Tr}(\mathbf{Q}_{\text{r-RESPA}})| = 2$, the two eigenvalues of $\mathbf{Q}_{\text{r-RESPA}}$ are identical, and equal to 1 or -1. Since stability requires the eigenvalue to be semi-simple in that case, it follows that the algorithm is stable if and only if $\mathbf{Q}_{\text{r-RESPA}} = \pm\mathbf{I}$, where $\mathbf{I}$ is the identity matrix. This leads to $h_1 = h_2 = 0$. Hence, the case $|\mathrm{Tr}(\mathbf{Q}_{\text{r-RESPA}})| = 2$ is always unstable.

Henceforth we shall assume that the stiff spring integrator is itself stable, i.e., that $h_1^2\Lambda_1 < 4$. The trace can then be expressed in terms of an invertible function $\theta\colon [0, 2/\sqrt{\Lambda_1}] \mapsto [0, \pi]$, $\theta(h_1)$, such that:

$$\cos\theta = 1 - \frac{h_1^2}{2}\Lambda_1, \quad \sin\theta = h_1\sqrt{\Lambda_1\left(1 - \frac{h_1^2}{4}\Lambda_1\right)}. \tag{3.3}$$

Denote:

$$\mathbf{G} = \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{\Lambda_1\left(1 - \frac{h_1^2}{4}\Lambda_1\right)} \end{bmatrix}, \text{ and } \mathbf{R}(\theta) = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}.$$

Then it can be shown that $\mathbf{Q}_{\text{fast}} = \mathbf{G}\mathbf{R}(\theta)\mathbf{G}^{-1}$, and so:

$$(\mathbf{Q}_{\text{fast}})^p = \mathbf{G}\mathbf{R}(p\,\theta)\mathbf{G}^{-1} \tag{3.4}$$

since $\mathbf{R}(\theta)$ is the rotation matrix. In this formulation an effective angular frequency can be defined as $\omega_{\text{eff}}(h_1) = \theta/h_1$ so the effective period is given by $T_{\text{eff}} = 2\pi/\omega_{\text{eff}} = 2\pi h_1/\theta$. Using this alternative form for $\mathbf{Q}_{\text{fast}}$ we find that

$$\mathrm{Tr}(\mathbf{Q}_{\text{r-RESPA}}) = 2\left[\cos(p\,\theta) - \alpha\sin(p\,\theta)\right] \tag{3.5}$$

where

$$\alpha = \frac{h_2\Lambda_2}{2\sqrt{\Lambda_1\left(1 - \frac{h_1^2}{4}\Lambda_1\right)}}. \tag{3.6}$$

The stability condition $|\mathrm{Tr}(\mathbf{Q}_{\text{r-RESPA}})| < 2$ can also be gleaned from constructing the shadow Hamiltonian. As noted before for VV, to construct a real-valued shadow Hamiltonian we will consider the integrator with propagator matrix $\mathbf{Q}^2_{\text{r-RESPA}}$. The shadow Hamiltonian for this integrator is

$$\tilde{H}_{\text{r-RESPA}}(q, p_q) = \begin{cases} \left(\frac{p_q^2}{2\gamma} + \frac{1}{2}\gamma\Lambda_1 q^2\right) \frac{\cos^{-1}\left(\frac{1}{2}\mathrm{Tr}(\mathbf{Q}_{\text{r-RESPA}})\right)}{h_2\sqrt{\Lambda_1}} & \text{if } |\mathrm{Tr}(\mathbf{Q}_{\text{r-RESPA}})| < 2 \\ s_{\mathrm{Tr}} \left[\frac{\sin(p\,\theta)}{2p\sin\theta}\right] p_q^2 & \text{if } |\mathrm{Tr}(\mathbf{Q}_{\text{r-RESPA}})| = 2 \\ s_{\mathrm{Tr}} \left(\frac{p_q^2}{2\gamma} - \frac{1}{2}\gamma\Lambda_1 q^2\right) \frac{\cosh^{-1}\left(\frac{1}{2}|\mathrm{Tr}(\mathbf{Q}_{\text{r-RESPA}})|\right)}{h_2\sqrt{\Lambda_1}} & \text{if } |\mathrm{Tr}(\mathbf{Q}_{\text{r-RESPA}})| > 2 \end{cases}$$

where p_q is the conjugate momentum of the spatial coordinate q and

$$s_{\mathrm{Tr}} = \mathrm{sgn}(\mathrm{Tr}(\mathbf{Q}_{\text{r-RESPA}}))$$
$$\gamma = \frac{1}{\sin(p\,\theta)}\sqrt{1 - \left(\frac{h_1}{2}\sqrt{\Lambda_1}\right)^2}\sqrt{\left|1 - \left(\frac{1}{2}\mathrm{Tr}(\mathbf{Q}_{\text{r-RESPA}})\right)^2\right|}.$$

Since the level sets of $\tilde{H}_{\text{r-RESPA}}$ are ellipses when $|\mathrm{Tr}(\mathbf{Q}_{\text{r-RESPA}})| < 2$ we conclude that r-RESPA is stable in this regime. However if $|\mathrm{Tr}(\mathbf{Q}_{\text{r-RESPA}})| = 2$ the level sets of $\tilde{H}_{\text{r-RESPA}}$ are now lines whereas for $|\mathrm{Tr}(\mathbf{Q}_{\text{r-RESPA}})| > 2$ the level sets are hyperbolas. In both cases these contours correspond to unstable trajectories. Therefore r-RESPA is unstable if $|\mathrm{Tr}(\mathbf{Q}_{\text{r-RESPA}})| \geq 2$.

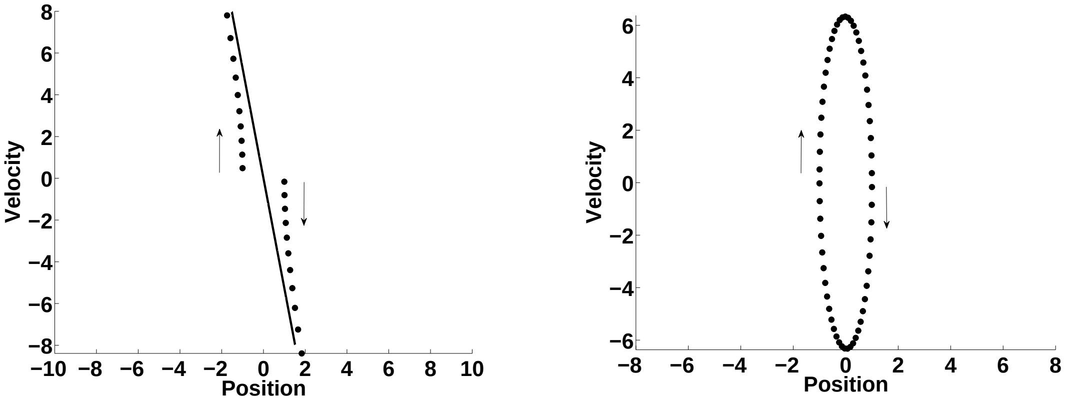
To determine what choice of time steps results in an unstable r-RESPA scheme, we start with the instability condition $|\mathrm{Tr}(\mathbf{Q}_{\text{r-RESPA}})| \geq 2$. This condition is satisfied when

$$\left|\cos\left(\frac{2\pi h_2}{T_{\text{eff}}} + \phi\right)\right| \geq \frac{1}{\sqrt{1+\alpha^2}}$$

where

$$\cos(\phi) = \frac{1}{\sqrt{1+\alpha^2}}, \quad \sin(\phi) = \frac{\alpha}{\sqrt{1+\alpha^2}}.$$

A few observations about these conditions are now appropriate. If $h_2 = mT_{\text{eff}}/2$ for some integer m, the algorithm is unstable, in agreement with the example discussed before. In fact, the instability appears for a range of values of h_2 near $mT_{\text{eff}}/2$. Notably, these values always lie on only one side of $mT_{\text{eff}}/2$; they are always slightly smaller. Any value slightly larger than $mT_{\text{eff}}/2$ leads to stable schemes, albeit with energy oscillations of very large amplitude. Looking at Fig. 3.2(a) taking h_2 to be slightly smaller than $T_{\text{eff}}/2$ gives an unstable scheme as shown by the diverging trajectory. On the other hand Fig. 3.2(b) shows that taking h_2 to be slightly larger than $T_{\text{eff}}/2$ gives an ellipsoidal trajectory and hence the algorithm is stable. However the high degree of stretching in the ellipse means that the energy exhibits large oscillations as a function of time.



(a) Time step h_2 slightly smaller than $T_{\text{eff}}/2$ (b) Time step h_2 slightly larger than $T_{\text{eff}}/2$

Figure 3.2: Phase plane diagrams for two choices of the time step h_2 near $T_{\text{eff}}/2$ with the trajectories sampled every h_2 and denoted by the dots. The trajectory on the left alternates between the second and fourth quadrants. The arrows point in the direction of increasing time. Here $\Lambda_1 = 0.9$, $\Lambda_2 = 0.1$, and $T_{\text{eff}}/2 \approx 3.3115$. Left: Taking the time step h_2 to be slightly smaller than $T_{\text{eff}}/2$ ($h_2 = 3.30$) the energy of the system grows unbounded as shown by the trajectory diverging from the origin. Both branches of the trajectory are approaching the eigenvector of $\mathbf{Q}_{\text{r-RESPA}}$ corresponding to the larger unstable eigenvalue (solid line). Right: Taking the time step h_2 to be slightly larger than $T_{\text{eff}}/2$ ($h_2 = 3.32$) the resulting trajectory is a closed loop. As a result the scheme is stable however the stretched ellipse implies that the energy exhibits large oscillations.

Using the analysis developed thus far, the width and amplitude of these resonances can be determined. When $\alpha \ll 1$ and $h_1\sqrt{\Lambda_1} \ll 1$, the resonance width around $h_2 = mT_{\text{eff}}/2$, or interval length of resonant time steps h_2, is found to be proportional to the ratio $\Lambda_2/\Lambda_1^{3/2}$

$$l \approx \frac{T_{\text{eff}}}{\pi}\alpha = \frac{h_2 T_{\text{eff}}}{2\pi\sqrt{\Lambda_1\left(1-\frac{h_1^2}{4}\Lambda_1\right)}}\Lambda_2 \approx m\pi\frac{\Lambda_2}{\Lambda_1^{3/2}} \tag{3.7}$$

where the first approximation uses $\tan^{-1}(x) \approx m\pi + x$ for x small and the second assumes that for h_1 small $T_{\text{eff}} \approx 2\pi/\sqrt{\Lambda_1}$. Therefore the resonance width decreases as the stiffness Λ_2 of the soft spring becomes softer relative to Λ_1. One implication of this is that resonances will persist even in the presence of a very soft spring but the probability of actually encountering them is low.

The resonance amplitude can be calculated by determining the magnitude of the largest eigenvalue. From the trace and determinant conditions the largest eigenvalue r_1 satisfies

$$|r_1| = \frac{1}{2}\left[|\text{Tr}(\mathbf{Q}_{\text{r-RESPA}})| + \sqrt{\text{Tr}(\mathbf{Q}_{\text{r-RESPA}})^2 - 4}\right].$$

The resonances occur for values of h_2 near integer multiples of $T_{\text{eff}}/2$:

$$\frac{2\pi h_2}{T_{\text{eff}}} = m\pi - \beta$$

where m is an integer and $0 < \beta < 2\alpha + O(\alpha^3)$. Assuming once again that α is small and that $h_1\sqrt{\Lambda_1} \ll 1$, we obtain the following approximation for $|r_1|$:

$$|r_1| \approx 1 + \beta\alpha - \frac{\beta^2}{2} + \sqrt{\frac{1}{3}\beta^4 - \beta^2 + \left(-\frac{4}{3}\beta^3 + 2\beta\right)\alpha + \beta^2\alpha^2}.$$

The maximum of $|r_1|$ is achieved near $\beta = \alpha$. With this, we get the following estimate for an upper bound on $|r_1|$:

$$\max_{\beta}|r_1| \approx 1 + \alpha + \frac{1}{2}\alpha^2 \approx 1 + \frac{m\,\pi}{2}\frac{\Lambda_2}{\Lambda_1}, \tag{3.8}$$

where we have only accounted for the linear term in α for the last expression. The amplitude of the resonance also decreases as the Λ_2 spring becomes softer relative to Λ_1. Note that the resonance amplitude is not maximum at $h_2 = mT_{\text{eff}}/2$, but near:

$$m\frac{T_{\text{eff}}}{2}\left(1 - \frac{1}{2}\frac{\Lambda_2}{\Lambda_1}\right).$$

3.3 Stability of AVI

We now turn our attention to the stability of the AVI algorithm. For two time steps h_1 and h_2 a propagation matrix $\mathbf{Q}_{\text{AVI}}$ can only be defined if there exists a synchronization point for the time integration of the two springs. This is the case when h_2/h_1 is a rational number p/q, where p and q are coprime integers. The two potentials then become synchronous at time $t = qh_2 = ph_1$. We will next prove a sufficient condition for the stability of the AVI algorithm, and numerically investigate the appearance of instabilities when the sufficient conditions are not satisfied.

Similarly to the study in Section 3.2, we need to calculate $\text{Tr}(\mathbf{Q}_{\text{AVI}})$. An exact equation for the trace can be found analytically for certain p and q (using a symbolic manipulation package for example). However, an equation valid for all p and q is obtained when a linearization in the variable Λ_2 is performed:

$$\text{Tr}(\mathbf{Q}_{\text{AVI}}) \approx 2\big[\cos(p\,\theta) - \alpha_q \sin(p\,\theta)\big] \tag{3.9}$$

where

$$\alpha_q = \frac{h_2\Lambda_2(q - \frac{q^2-1}{q}\frac{h_1^2}{6}\Lambda_1)}{2\sqrt{\Lambda_1(1 - \frac{h_1^2}{4}\Lambda_1)}}.$$

In fact, we provide an error bound for this approximation in the following proposition.

Proposition 1 *Assume that the fast integrator is stable, namely, $h_1^2\Lambda_1 < 4$. If time steps $h_1 < h_2$ satisfy that $h_2/h_1 = p/q$, for some p and q coprime integers, then the* *following inequality holds:*

$$\boxed{\left|\mathrm{Tr}(\mathbf{Q}_{\mathrm{AVI}}) - 2\big[\cos(p\,\theta) - \alpha_q \sin(p\,\theta)\big]\right| < (2q\alpha_1)^2 \left(1 + 2\alpha_1^2 + 2\alpha_1\sqrt{1+\alpha_1^2}\right)^{\frac{q-2}{2}}} \tag{3.10}$$

where $\theta = h_1\omega_{\mathrm{eff}}$ *is defined in equation (3.3).*

Before proving this result, let's consider some of its implications. Notice first that the analysis in Section 3.2 (see Eqs. (3.5) and (3.6)) corresponds to the case $q = 1$. In that case, the trace is exactly linear in Λ_2 and Eq. (3.9) is exact. Next, Eq. (3.10) provides a sufficient condition for stability, namely, the AVI algorithm is stable if

$$\boxed{2\left|\cos(p\,\theta) - \alpha_q \sin(p\,\theta)\right| + (2q\alpha_1)^2 \left(1 + 2\alpha_1^2 + 2\alpha_1\sqrt{1+\alpha_1^2}\right)^{\frac{q-2}{2}} < 2.} \tag{3.11}$$

To better illustrate when instabilities may be found, we write

$$\cos(p\,\theta) - \alpha_q \sin(p\,\theta) = \sqrt{1+\alpha_q^2}\ \cos(p\,\theta + \phi)$$

with

$$\cos\phi = 1/\sqrt{1+\alpha_q^2} \qquad \text{and} \qquad \sin\phi = \alpha_q/\sqrt{1+\alpha_q^2}, \tag{3.12}$$

from where it follows that

$$\mathrm{Tr}(\mathbf{Q}_{\mathrm{AVI}}) \approx 2\sqrt{1+\alpha_q^2}\ \cos(p\,\theta + \phi),$$

provided that the right-hand side in Eq. (3.10) is small enough. In these circumstances, a sufficient condition for the integrator to *be stable* is for $p\,\theta + \phi = qh_2 w_{\mathrm{eff}} + \phi$ to be sufficiently away from $m\pi$. This is more precisely stated in the following corollary:

Corollary 1 *Assume that the fast integrator is stable, namely,* $h_1^2\Lambda_1 < 4$*. Then, for any* $\varepsilon > 0$ *there exists* $\eta > 0$ *such that if* $|p\,\theta + \phi - m\pi| > \varepsilon$ *for all* $m \in \mathbb{Z}$ *and*

$qh_2\Lambda_2/\sqrt{\Lambda_1} < \eta$ then

$$|\mathrm{Tr}(\mathbf{Q}_{\mathrm{AVI}})| < 2,$$

and the AVI integrator is stable.

Proof. Observe first that the condition $|p\,\theta + \phi - m\pi| > \varepsilon$ for all $m \in \mathbb{Z}$ implies that $|\cos(p\,\theta + \phi)\,| < |\cos(\varepsilon)|$. Next, since $h_1^2\Lambda_1 < 4$, we have that $\alpha_1, \alpha_q \in \mathbb{R}$ and that $\alpha_q = O(q\alpha_1)$. Finally, notice that if $0 < qh_2\Lambda_2/\sqrt{\Lambda_1} < \eta$ then

$$0 < q\alpha_1 < \frac{\eta}{2\sqrt{1 - \frac{h_1^2}{4}\Lambda_1}} < \hat{\eta}, \tag{3.13}$$

and the right hand side of Eq. (3.10) satisfies

$$\begin{aligned}(2q\alpha_1)^2\left(1 + 2\alpha_1^2 + 2\alpha_1\sqrt{1+\alpha_1^2}\right)^{\frac{q-2}{2}} &< (2\hat{\eta})^2 \sup_{0\le\alpha_1\le\hat{\eta}} \left(1 + 2\alpha_1^2 + 2\alpha_1\sqrt{1+\alpha_1^2}\right)^{\frac{\hat{\eta}}{2\alpha_1}-1} \\ &< (2\hat{\eta})^2\exp(\hat{\eta}).\end{aligned} \tag{3.14}$$

It is then always possible to choose $\eta > 0$ such that

$$\begin{aligned}|\mathrm{Tr}(\mathbf{Q}_{\mathrm{AVI}})| &< 2|\cos(\varepsilon)|\sqrt{1+\alpha_q^2} + (2q\alpha_1)^2\left(1 + 2\alpha_1^2 + 2\alpha_1\sqrt{1+\alpha_1^2}\right)^{\frac{q-2}{2}} \\ &< |2\cos(\varepsilon)|\sqrt{1+\alpha_q^2} + (2\hat{\eta})^2\exp(\hat{\eta}) \\ &< 2\end{aligned}$$

□

This corollary generalizes the r-RESPA case ($q = 1$) for $qh_2\Lambda_2/\sqrt{\Lambda_1} \ll 1$, since in this case ϕ is also small and hence

$$\boxed{\text{the system is stable whenever } ph_1 = qh_2 \text{ is away from } mT_{\mathrm{eff}}/2.}$$

3.3.1 Proof of Proposition 1

We begin by constructing the propagation matrix $\mathbf{Q}_{\mathrm{AVI}}$ over the time interval $t = 0$ to $t = q\,h_2 = p\,h_1$ as a composition of multiple elementary matrices. A simple expression of the resulting matrix product is in most cases difficult to obtain, so the key step in the proof is to perform a Taylor expansion for the trace of $\mathbf{Q}_{\mathrm{AVI}}$ in terms of Λ_2, which leads to several simplifications. To this end,

$$\mathrm{Tr}(\mathbf{Q}_{\mathrm{AVI}})(\Lambda_2) = \mathrm{Tr}(\mathbf{Q}_{\mathrm{AVI}})(0) + \Lambda_2 \left.\frac{d\mathrm{Tr}(\mathbf{Q}_{\mathrm{AVI}})}{d\Lambda_2}\right|_{\Lambda_2=0} + \frac{\Lambda_2^2}{2} \left.\frac{d^2\mathrm{Tr}(\mathbf{Q}_{\mathrm{AVI}})}{d\Lambda_2^2}\right|_{\Lambda_2^*}$$

This is exact for some $0 < \Lambda_2^* < \Lambda_2$. We will show that:

$$\mathrm{Tr}(\mathbf{Q}_{\mathrm{AVI}})(0) + \Lambda_2 \left.\frac{d\mathrm{Tr}(\mathbf{Q}_{\mathrm{AVI}})}{d\Lambda_2}\right|_{\Lambda_2=0} = 2\left(\cos(p\,\theta) - \alpha_q \sin(p\,\theta)\right) \tag{3.15}$$

$$\left|\frac{\Lambda_2^2}{2} \left.\frac{d^2\mathrm{Tr}(\mathbf{Q}_{\mathrm{AVI}})}{d\Lambda_2^2}\right|_{\Lambda_2^*}\right| < (2q\alpha_1)^2 \left(1 + 2\alpha_1^2 + 2\alpha_1\sqrt{1+\alpha_1^2}\right)^{\frac{q-2}{2}} \tag{3.16}$$

These two equations prove our result.

We begin by defining a few matrices which are the building blocks of AVI:

$$\mathbf{V}_{\mathrm{s}} = \begin{bmatrix} 1 & 0 \\ -\frac{h_2}{2}\Lambda_2 & 1 \end{bmatrix}, \quad \mathbf{V}_{\mathrm{f}} = \begin{bmatrix} 1 & 0 \\ -\frac{h_1}{2}\Lambda_1 & 1 \end{bmatrix}, \quad \mathbf{U}_i = \begin{bmatrix} 1 & \frac{i}{q}h_1 \\ 0 & 1 \end{bmatrix}$$

$$\mathbf{Q}_{\mathrm{f}} = \mathbf{V}_{\mathrm{f}}\mathbf{U}_q\mathbf{V}_{\mathrm{f}} = \begin{bmatrix} 1 - \frac{h_1^2}{2}\Lambda_1 & h_1 \\ -h_1\Lambda_1\left(1 - \frac{h_1^2}{4}\Lambda_1\right) & 1 - \frac{h_1^2}{2}\Lambda_1 \end{bmatrix}$$

$$\mathbf{Q}_{\mathrm{s}}(m) = \mathbf{Q}_{\mathrm{f}}^{-1}[\mathbf{V}_{\mathrm{f}}\mathbf{U}_{q-m}\mathbf{V}_{\mathrm{s}}\mathbf{V}_{\mathrm{s}}\mathbf{U}_m\mathbf{V}_{\mathrm{f}}]$$

Some of these definitions have been previously introduced. Matrices $\mathbf{V}_{\mathrm{s}}$ and $\mathbf{V}_{\mathrm{f}}$ correspond to kicks by the soft and the stiff springs, respectively, while matrix $\mathbf{U}_i$ represents a drift for a time interval of length ih_1/q. The matrix $\mathbf{Q}_{\mathrm{f}}$ is the propagation matrix for a complete time step of the fast spring. Similarly, $\mathbf{Q}_{\mathrm{f}}\,\mathbf{Q}_{\mathrm{s}}(m)$ represents a kick-and-drift step with the stiff spring, followed by a kick by the soft spring, and a drift-and-kick step with the stiff spring. The drift time m/qh_1 ($\mathbf{U}_m$) is required to reach the time at which the soft spring kicks. After the soft kick, the system drifts for a time $(1 - m/q)h_1$ ($\mathbf{U}_{q-m}$) to reach the next time step for the stiff spring. The presence of $\mathbf{Q}_\mathrm{f}^{-1}$ in the definition of $\mathbf{Q}_\mathrm{s}$ was added for later convenience.

Let us introduce the sequence $m_i = i\,p \pmod q$, and:

$$k_i = \left\lfloor \frac{ip}{q} \right\rfloor - \left\lfloor \frac{(i-1)p}{q} \right\rfloor,\ 1 \le i \le q,$$

where $\lfloor x \rfloor$ stands for the largest integer smaller than x. The value of k_i is the number of time steps the stiff spring needs to perform between time steps $i-1$ and i of the soft spring. Notice that this is exact because of the definition of $\mathbf{Q}_\mathrm{s}$ with the factor $\mathbf{Q}_\mathrm{f}^{-1}$. It then naturally follows that

$$\sum_{i=1}^{q} k_i = p. \tag{3.17}$$

The propagation matrix $\mathbf{Q}_\mathrm{AVI}$ is then given by:

$$\boxed{\mathbf{Q}_\mathrm{AVI} = \mathbf{V}_\mathrm{s}(\mathbf{Q}_\mathrm{f})^{k_q}\mathbf{Q}_\mathrm{s}(m_{q-1})(\mathbf{Q}_\mathrm{f})^{k_{q-1}}\cdots\mathbf{Q}_\mathrm{s}(m_1)(\mathbf{Q}_\mathrm{f})^{k_1}\mathbf{V}_\mathrm{s}.}$$

This product leads to a complicated expression for arbitrary values of Λ_2, but it leads to a surprisingly simple one in the limit of very small Λ_2. Since $\mathbf{Q}_\mathrm{AVI}$ is an infinitely smooth function of Λ_2, we can apply Taylor's theorem. The constant term is

$$\mathbf{Q}_\mathrm{AVI}\Big|_{\Lambda_2=0} = (\mathbf{Q}_\mathrm{f})^p.$$

The first derivative is equal to:

$$\frac{\partial \mathbf{Q}_\mathrm{AVI}}{\partial \Lambda_2}\Big|_{\Lambda_2=0} = \frac{\partial \mathbf{V}_\mathrm{s}}{\partial \Lambda_2}(\mathbf{Q}_\mathrm{f})^p + (\mathbf{Q}_\mathrm{f})^p\frac{\partial \mathbf{V}_\mathrm{s}}{\partial \Lambda_2} + \sum_{i=1}^{q-1}(\mathbf{Q}_\mathrm{f})^{p-\lfloor \frac{ip}{q}\rfloor}\frac{\partial \mathbf{Q}_\mathrm{s}(m_i)}{\partial \Lambda_2}(\mathbf{Q}_\mathrm{f})^{\lfloor \frac{ip}{q}\rfloor}$$

where the derivatives are evaluated at $\Lambda_2 = 0$. We can calculate the linear part of the trace:

$$\mathrm{Tr}(\mathbf{Q}_{\mathrm{AVI}}) = \mathrm{Tr}((\mathbf{Q}_{\mathrm{f}})^p) + \Lambda_2 \,\mathrm{Tr}\left(2(\mathbf{Q}_{\mathrm{f}})^p \frac{\partial \mathbf{V}_{\mathrm{s}}}{\partial \Lambda_2} + (\mathbf{Q}_{\mathrm{f}})^p \sum_{i=1}^{q-1} \frac{\partial \mathbf{Q}_{\mathrm{s}}(m_i)}{\partial \Lambda_2} \right) \Bigg|_{\Lambda_2=0} + O(\Lambda_2^2)$$

using the fact that $\mathrm{Tr}(\mathbf{AB}) = \mathrm{Tr}(\mathbf{BA})$. Since m_i is a permutation of $1, \cdots, q-1$, we also have:

$$\mathrm{Tr}(\mathbf{Q}_{\mathrm{AVI}}) = \mathrm{Tr}((\mathbf{Q}_{\mathrm{f}})^p) + \Lambda_2 \,\mathrm{Tr}\left(2(\mathbf{Q}_{\mathrm{f}})^p \frac{\partial \mathbf{V}_{\mathrm{s}}}{\partial \Lambda_2} + (\mathbf{Q}_{\mathrm{f}})^p \sum_{i=1}^{q-1} \frac{\partial \mathbf{Q}_{\mathrm{s}}(i)}{\partial \Lambda_2} \right) \Bigg|_{\Lambda_2=0} + O(\Lambda_2^2) \tag{3.18}$$

where i has been substituted instead of m_i in $\mathbf{Q}_{\mathrm{s}}$. The term $\partial \mathbf{Q}_{\mathrm{s}}(i)/\partial \Lambda_2$ at $\Lambda_2 = 0$ is a quadratic function of i which we write as:

$$\frac{\partial \mathbf{Q}_{\mathrm{s}}(i)}{\partial \Lambda_2} \Bigg|_{\Lambda_2=0} = \mathbf{A}_0 + \mathbf{A}_1 i + \mathbf{A}_2 i^2$$

The coefficients can be obtained from the definition of $\mathbf{Q}_{\mathrm{s}}(i)$:

$$\mathbf{A}_0 = -h_2 \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \quad \mathbf{A}_1 = \frac{h_1 h_2}{q} \begin{bmatrix} 1 & 0 \\ h_1 \Lambda_1 & -1 \end{bmatrix}, \quad \mathbf{A}_2 = \frac{h_1^2 h_2}{q^2} \begin{bmatrix} -\frac{h_1 \Lambda_1}{2} & 1 \\ -\frac{h_1^2 \Lambda_1^2}{4} & \frac{h_1 \Lambda_1}{2} \end{bmatrix}$$

The sum can therefore be computed analytically:

$$\sum_{i=1}^{q-1} \frac{\partial \mathbf{Q}_{\mathrm{s}}(i)}{\partial \Lambda_2} \Bigg|_{\Lambda_2=0} = (q-1)\,\mathbf{A}_0 + \frac{(q-1)q}{2}\,\mathbf{A}_1 + \frac{(q-1)q(2q-1)}{6}\,\mathbf{A}_2,$$

which, together with the value of $(\mathbf{Q}_{\mathrm{f}})^p$ in Eq. (3.4), enables the direct computation of the second term in Eq. (3.18):

$$\Lambda_2 \,\mathrm{Tr}\left(2(\mathbf{Q}_{\mathrm{f}})^p \frac{\partial \mathbf{V}_{\mathrm{s}}}{\partial \Lambda_2} + (\mathbf{Q}_{\mathrm{f}})^p \sum_{i=1}^{q-1} \frac{\partial \mathbf{Q}_{\mathrm{s}}(i)}{\partial \Lambda_2} \right) \Bigg|_{\Lambda_2=0} = -\frac{h_2 \Lambda_2 \left(q - \left(\frac{q^2-1}{q} \right) \frac{h_1^2}{6} \Lambda_1 \right)}{\sqrt{\Lambda_1 (1 - \frac{h_1^2}{4} \Lambda_1)}} \sin(p\theta). \tag{3.19}$$

The first term in the same equation follows as

$$\mathrm{Tr}((\mathbf{Q}_{\mathrm{f}})^{p}) = \mathrm{Tr}(\mathbf{G}\mathbf{R}(p\,\theta)\mathbf{G}^{-1}) = \mathrm{Tr}(\mathbf{R}(p\,\theta)) = 2\cos(p\theta),$$

Together with Eqs. (3.19) and (3.18), this proves Eq. (3.15).

We now establish a bound on the second derivative of $\mathrm{Tr}(\mathbf{Q}_{\mathrm{AVI}})$. The first derivative at an arbitrary Λ_2 is given by:

$$\begin{aligned}\frac{\partial \mathbf{Q}_{\mathrm{AVI}}}{\partial \Lambda_2} &= \frac{\partial \mathbf{V}_{\mathrm{s}}}{\partial \Lambda_2}(\mathbf{Q}_{\mathrm{f}})^{k_q}\mathbf{Q}_{\mathrm{s}}(m_{q-1})(\mathbf{Q}_{\mathrm{f}})^{k_{q-1}}\cdots\mathbf{Q}_{\mathrm{s}}(m_1)(\mathbf{Q}_{\mathrm{f}})^{k_1}\mathbf{V}_{\mathrm{s}} \\ &+ \mathbf{V}_{\mathrm{s}}(\mathbf{Q}_{\mathrm{f}})^{k_q}\mathbf{Q}_{\mathrm{s}}(m_{q-1})(\mathbf{Q}_{\mathrm{f}})^{k_{q-1}}\cdots\mathbf{Q}_{\mathrm{s}}(m_1)(\mathbf{Q}_{\mathrm{f}})^{k_1}\frac{\partial \mathbf{V}_{\mathrm{s}}}{\partial \Lambda_2} \\ &+ \mathbf{V}_{\mathrm{s}}\left[\sum_{i=1}^{q-1}(\mathbf{Q}_{\mathrm{f}})^{k_q}\mathbf{Q}_{\mathrm{s}}(m_{q-1})(\mathbf{Q}_{\mathrm{f}})^{k_{q-1}}\cdots\frac{\partial \mathbf{Q}_{\mathrm{s}}}{\partial \Lambda_2}(m_i)\cdots\mathbf{Q}_{\mathrm{s}}(m_1)(\mathbf{Q}_{\mathrm{f}})^{k_1}\right]\mathbf{V}_{\mathrm{s}}\end{aligned}$$

Using the facts that $\partial^2\mathbf{V}_{\mathrm{s}}/\partial\Lambda_2^2 = 0$ and $\partial^2\mathbf{Q}_{\mathrm{s}}(m)/\partial\Lambda_2^2 = 0$, we have

$$\begin{aligned}\frac{\partial^2 \mathbf{Q}_{\mathrm{AVI}}}{\partial \Lambda_2^2} &= 2\frac{\partial \mathbf{V}_{\mathrm{s}}}{\partial \Lambda_2}(\mathbf{Q}_{\mathrm{f}})^{k_q}\mathbf{Q}_{\mathrm{s}}(m_{q-1})(\mathbf{Q}_{\mathrm{f}})^{k_{q-1}}\cdots\mathbf{Q}_{\mathrm{s}}(m_1)(\mathbf{Q}_{\mathrm{f}})^{k_1}\frac{\partial \mathbf{V}_{\mathrm{s}}}{\partial \Lambda_2} \\ &+ 2\frac{\partial \mathbf{V}_{\mathrm{s}}}{\partial \Lambda_2}\left[\sum_{i=1}^{q-1}(\mathbf{Q}_{\mathrm{f}})^{k_q}\mathbf{Q}_{\mathrm{s}}(m_{q-1})(\mathbf{Q}_{\mathrm{f}})^{k_{q-1}}\cdots\frac{\partial \mathbf{Q}_{\mathrm{s}}}{\partial \Lambda_2}(m_i)\cdots\mathbf{Q}_{\mathrm{s}}(m_1)(\mathbf{Q}_{\mathrm{f}})^{k_1}\right]\mathbf{V}_{\mathrm{s}} \\ &+ 2\mathbf{V}_{\mathrm{s}}\left[\sum_{i=1}^{q-1}(\mathbf{Q}_{\mathrm{f}})^{k_q}\mathbf{Q}_{\mathrm{s}}(m_{q-1})(\mathbf{Q}_{\mathrm{f}})^{k_{q-1}}\cdots\frac{\partial \mathbf{Q}_{\mathrm{s}}}{\partial \Lambda_2}(m_i)\cdots\mathbf{Q}_{\mathrm{s}}(m_1)(\mathbf{Q}_{\mathrm{f}})^{k_1}\right]\frac{\partial \mathbf{V}_{\mathrm{s}}}{\partial \Lambda_2} \\ &+ 2\mathbf{V}_{\mathrm{s}}\left[\sum_{\substack{i,j=1\\ i<j}}^{q-1}(\mathbf{Q}_{\mathrm{f}})^{k_q}\mathbf{Q}_{\mathrm{s}}(m_{q-1})(\mathbf{Q}_{\mathrm{f}})^{k_{q-1}}\cdots\frac{\partial \mathbf{Q}_{\mathrm{s}}}{\partial \Lambda_2}(m_j)\cdots\frac{\partial \mathbf{Q}_{\mathrm{s}}}{\partial \Lambda_2}(m_i)\cdots\mathbf{Q}_{\mathrm{s}}(m_1)(\mathbf{Q}_{\mathrm{f}})^{k_1}\right]\mathbf{V}_{\mathrm{s}}\end{aligned} \tag{3.20}$$

In order to bound the second derivative, we need to recall some properties of matrix norms. Let $\mathbf{Q}$ be an $n \times n$ matrix, $n \in \mathbb{N}$, then

$$\|\mathbf{Q}\|_{\mathrm{E}}^2 = \mathrm{Tr}(\mathbf{Q}^{\mathsf{T}}\mathbf{Q}) \qquad \|\mathbf{Q}\|_2^2 = \sup_{\vec{x}\neq\vec{0}}\frac{\vec{x}^{\mathsf{T}}\mathbf{Q}^{\mathsf{T}}\mathbf{Q}\vec{x}}{\vec{x}^{\mathsf{T}}\vec{x}},$$

and it holds that $\|\mathbf{Q}\|_{\mathrm{E}} = n^{1/2}\|\mathbf{Q}\|_2$. Additionally, for any two $n \times n$ matrices $\mathbf{P}$ and

$\mathbf{Q}$, it holds that

$$\|\mathbf{PQ}\| \leq \|\mathbf{P}\|\,\|\mathbf{Q}\|,$$

in any of the two norms. Finally, by Cauchy-Schwartz inequality we have

$$|\mathrm{Tr}(\mathbf{AB})| \leq \|\mathbf{A}\|_{\mathrm{E}}\|\mathbf{B}\|_{\mathrm{E}} \leq n\|\mathbf{A}\|_2\|\mathbf{B}\|_2 \tag{3.21}$$

We simplify the notations for clarity:

$$\begin{aligned}
\mathbf{Q}_i &= \mathbf{G}^{-1}\mathbf{Q}_{\mathrm{s}}(m_i)\mathbf{G} \\
\mathbf{R}_i &= \mathbf{R}(k_i\theta),
\end{aligned}$$

which transforms Eq. (3.20) into

$$\begin{aligned}
\frac{\partial^2\mathbf{Q}_{\mathrm{AVI}}}{\partial\Lambda_2^2} &= 2\,\frac{\partial\mathbf{V}_{\mathrm{s}}}{\partial\Lambda_2}\mathbf{G}\mathbf{R}_q\mathbf{Q}_{q-1}\mathbf{R}_{q-1}\cdots\mathbf{Q}_1\mathbf{R}_1\mathbf{G}^{-1}\frac{\partial\mathbf{V}_{\mathrm{s}}}{\partial\Lambda_2} \\
&+ 2\,\frac{\partial\mathbf{V}_{\mathrm{s}}}{\partial\Lambda_2}\mathbf{G}\left[\sum_{i=1}^{q-1}\mathbf{R}_{k_q}\mathbf{Q}_{q-1}\mathbf{R}_{k_{q-1}}\cdots\frac{\partial\mathbf{Q}_i}{\partial\Lambda_2}\cdots\mathbf{Q}_1\mathbf{R}_{k_1}\right]\mathbf{G}^{-1}\mathbf{V}_{\mathrm{s}} \\
&+ 2\,\mathbf{V}_{\mathrm{s}}\mathbf{G}\left[\sum_{i=1}^{q-1}\mathbf{R}_{k_q}\mathbf{Q}_{q-1}\mathbf{R}_{k_{q-1}}\cdots\frac{\partial\mathbf{Q}_i}{\partial\Lambda_2}\cdots\mathbf{Q}_1\mathbf{R}_{k_1}\right]\mathbf{G}^{-1}\frac{\partial\mathbf{V}_{\mathrm{s}}}{\partial\Lambda_2} \\
&+ 2\,\mathbf{V}_{\mathrm{s}}\mathbf{G}\Bigg[\sum_{\substack{i,j=1\\ i<j}}^{q-1}\mathbf{R}_{k_q}\mathbf{Q}_{q-1}\mathbf{R}_{k_{q-1}}\cdots\frac{\partial\mathbf{Q}_j}{\partial\Lambda_2}\cdots\frac{\partial\mathbf{Q}_i}{\partial\Lambda_2}\cdots\mathbf{Q}_1\mathbf{R}_{k_1}\Bigg]\mathbf{G}^{-1}\mathbf{V}_{\mathrm{s}}
\end{aligned}$$

We now simply denote $\|\cdot\|$ instead of $\|\cdot\|_2$. Noticing that $\|\mathbf{R}_i\| = 1$, we have that

$$
\begin{aligned}
&\frac{1}{4}\left|\mathrm{Tr}\left(\frac{\partial^2 \mathbf{Q}_{\mathrm{AVI}}}{\partial \Lambda_2^2}\right)\right| \leq \\
&\quad \leq \left\|\mathbf{G}^{-1}\frac{\partial \mathbf{V}_{\mathrm{s}}}{\partial \Lambda_2}\frac{\partial \mathbf{V}_{\mathrm{s}}}{\partial \Lambda_2}\mathbf{G}\right\| \; \|\mathbf{Q}_1\| \cdots \|\mathbf{Q}_{q-1}\| \\
&\qquad + \left(\left\|\mathbf{G}^{-1}\mathbf{V}_{\mathrm{s}}\frac{\partial \mathbf{V}_{\mathrm{s}}}{\partial \Lambda_2}\mathbf{G}\right\| + \left\|\mathbf{G}^{-1}\frac{\partial \mathbf{V}_{\mathrm{s}}}{\partial \Lambda_2}\mathbf{V}_{\mathrm{s}}\mathbf{G}\right\|\right) \\
&\qquad\quad \left[\sum_{i=1}^{q-1} \|\mathbf{Q}_1\| \cdots \|\mathbf{Q}_{i-1}\| \; \left\|\frac{\partial \mathbf{Q}_i}{\partial \Lambda_2}\right\| \; \|\mathbf{Q}_{i+1}\| \cdots \|\mathbf{Q}_{q-1}\|\right] \qquad (3.22) \\
&\qquad + \|\mathbf{G}^{-1}\mathbf{V}_{\mathrm{s}}\mathbf{V}_{\mathrm{s}}\mathbf{G}\| \; \left[\sum_{\substack{i,j=1 \\ i<j}}^{q-1} \|\mathbf{Q}_1\| \cdots \|\mathbf{Q}_{i-1}\| \; \left\|\frac{\partial \mathbf{Q}_i}{\partial \Lambda_2}\right\| \; \|\mathbf{Q}_{i+1}\| \cdots \right. \\
&\qquad\quad \left. \cdots \|\mathbf{Q}_{j-1}\| \; \left\|\frac{\partial \mathbf{Q}_j}{\partial \Lambda_2}\right\| \; \|\mathbf{Q}_{j+1}\| \cdots \|\mathbf{Q}_{q-1}\|\right].
\end{aligned}
$$

This equation is obtained by a simple application of Eq. (3.21). It can then be verified by direct calculation that

$$
\left\|\mathbf{G}^{-1}\frac{\partial \mathbf{V}_{\mathrm{s}}}{\partial \Lambda_2}\frac{\partial \mathbf{V}_{\mathrm{s}}}{\partial \Lambda_2}\mathbf{G}\right\| = 0 \qquad (3.23)
$$

$$
\left\|\mathbf{G}^{-1}\mathbf{V}_{\mathrm{s}}\frac{\partial \mathbf{V}_{\mathrm{s}}}{\partial \Lambda_2}\mathbf{G}\right\| = \left\|\mathbf{G}^{-1}\frac{\partial \mathbf{V}_{\mathrm{s}}}{\partial \Lambda_2}\mathbf{V}_{\mathrm{s}}\mathbf{G}\right\| = \frac{\alpha_1}{\Lambda_2} \qquad (3.24)
$$

$$
\|\mathbf{G}^{-1}\mathbf{V}_{\mathrm{s}}\mathbf{V}_{\mathrm{s}}\mathbf{G}\| = \left(1 + 2\alpha_1^2 + 2\alpha_1\sqrt{1+\alpha_1^2}\right)^{\frac{1}{2}} \qquad (3.25)
$$

$$
\left\|\frac{\partial \mathbf{Q}_i}{\partial \Lambda_2}\right\| \leq \frac{2\alpha_1}{\Lambda_2} \qquad (3.26)
$$

$$
\|\mathbf{Q}_i\| \leq \left(1 + 2\alpha_1^2 + 2\alpha_1\sqrt{1+\alpha_1^2}\right)^{\frac{1}{2}}. \qquad (3.27)
$$

The last two inequalities use the fact that

$$
|q^2 - i(q-i)h_1^2\Lambda_1| < q^2, \quad \text{for } 0 < h_1^2\Lambda_1 < 4 \text{ and } 1 \leq i \leq q-1.
$$

Applying these results in Eq. (3.22), we get

$$\begin{aligned}\frac{\Lambda_2^2}{2}\left|\mathrm{Tr}\left(\frac{\partial^2\mathbf{Q}_{\mathrm{AVI}}}{\partial\Lambda_2^2}\right)\right|\Bigg|_{\Lambda_2^*} &\leq 4q(q-1)\alpha_1^2\left(1+2\alpha_1^2+2\alpha_1\sqrt{1+\alpha_1^2}\right)^{\frac{q-2}{2}}\\ &< (2q\alpha_1)^2\left(1+2\alpha_1^2+2\alpha_1\sqrt{1+\alpha_1^2}\right)^{\frac{q-2}{2}},\end{aligned}$$

for any Λ_2^* such that $0 \leq \Lambda_2^* \leq \Lambda_2$, which proves Eq. (3.16).

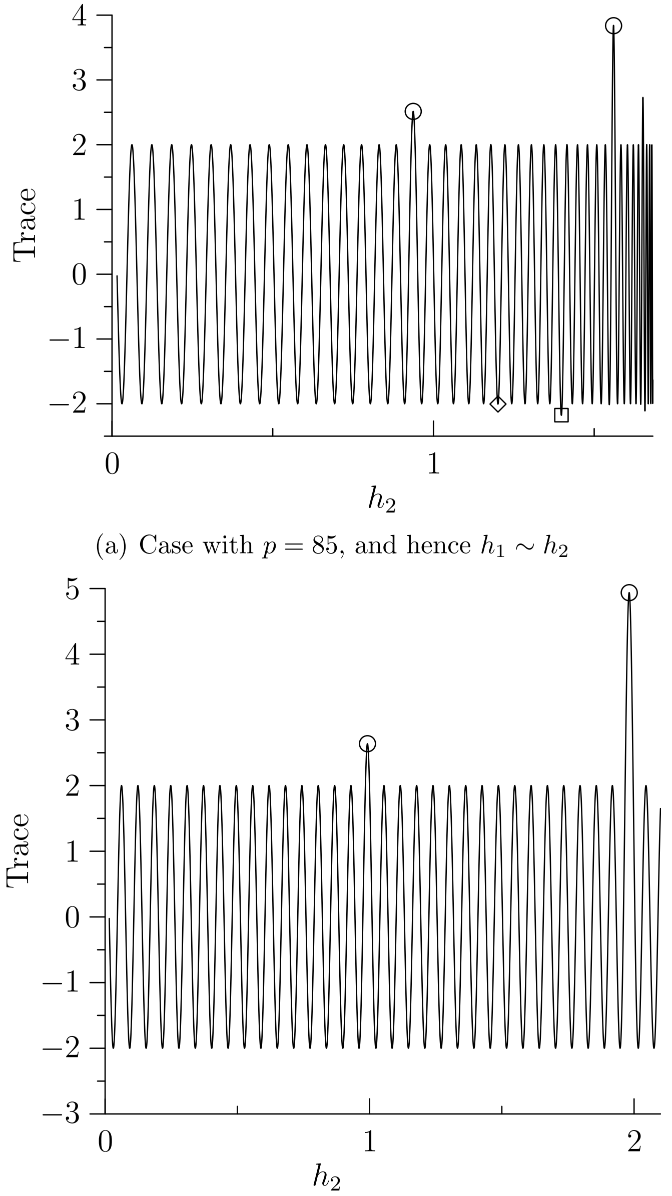
□

3.3.2 AVI and r-RESPA resonances

In the forthcoming sections we denote the effective half-period $T_{\mathrm{eff}}(h_1)/2$ by $T_{1/2}(h_1)$. In Sections 3.3.2 and 3.3.3, all numerical results were obtained using extended precision arithmetic (53 digits were typically used), and we will define a *resonant interval* as an interval in the real line that contains all unstable time step values.

Given p and q, consider the problem of finding resonant points (h_1, h_2) along the line $h_2 = \frac{p}{q}h_1$. Our previous analysis predicts that the resonant points are approximately located at $h_2 = \frac{m}{q}T_{1/2}$ for $qh_2\Lambda_2/\sqrt{\Lambda_1} \ll 1$. The behavior for larger values of $qh_2\Lambda_2/\sqrt{\Lambda_1}$ was investigated numerically and presented next.

A typical result is illustrated in Fig. 3.3, which shows the value of $\mathrm{Tr}(\mathbf{Q}_{\mathrm{AVI}})$ as a function of h_2. We note that: a) its value oscillates between -2 and 2 with a frequency close to $p\theta/h_2 \approx qw_{\mathrm{eff}}$ and b) an exhaustive examination of each local extremum reveals that the value of the trace at each one of them is a resonant point. This leads to resonances located at approximately $h_2 = mT_{1/2}(h_1)/q$, as before. Other numerical tests consistently displayed the same behavior as well. However, the presence of these two features in all examples does not follow from the result of Proposition 1. We conjecture that both characteristics are generally true, as expressed in the following statement, which remains to be proved:



(a) Case with $p = 85$, and hence $h_1 \sim h_2$

(b) Case with $p = 1025$, and hence $h_1 \ll h_2$

Figure 3.3: Trace of $\mathbf{Q}_{\text{AVI}}$ as a function of h_2, when $h_2 = ph_1/q$ for given values of p and q. Two different examples are shown, which differ in the value of p, but both adopt $q = 32$, $\Lambda_1 = \pi^2$, $\Lambda_2 = \pi^2/64$. When the value of the trace is outside of the interval $(-2, 2)$, the integrator is unstable. Zooming in on the regions where the trace is near 2 or -2 shows that, in each instance, there is an interval of instability. However these instabilities are weak except when h_2 is near a multiple of $T_{1/2}$: these are depicted with circles. The main difference between the two cases is that the top plot shows additional large resonances besides the multiples of $T_{1/2}$. These are depicted by a square and a diamond.

Let $t_{p,q}(h_2)$ denote the value of $\mathrm{Tr}(\mathbf{Q}_{\mathrm{AVI}})$ evaluated at $(h_1, h_2) = (qh_2/p, h_2)$, for any $p > q$ coprime integers. Then, there exists a constant C independent of p and q such that for any $h_2 \in (0, 2p/(q\sqrt{\Lambda_1}))$ there exists an extremizer h_2^r of $t_{p,q}(h_2)$ that satisfies

$$h_2 \leq h_2^r < h_2 + \frac{C}{q\sqrt{\Lambda_1}}. \tag{3.28}$$

Additionally, all local extremizers are resonant points.

Eq. (3.28) is motivated by the previous qualitative observation that the function $t_{p,q}(h_2)$ oscillates as h_2 changes with a frequency close to $qw_{\mathrm{eff}}(qh_2/p)$. It then easily follows that the corresponding approximate period is bounded as

$$\frac{2\pi}{qw_{\mathrm{eff}}(h_1)} = 2\pi \frac{h_1}{q\theta(h_1)} \leq \frac{2\pi}{q\sqrt{\Lambda_1}}. \tag{3.29}$$

The last inequality follows after noticing that

$$\theta(h_1) \geq \sqrt{\Lambda_1} h_1, \tag{3.30}$$

for $h_1 \in [0, \pi]$. The restriction $0 < h_2 < 2p/(q\sqrt{\Lambda_1})$ guarantees that only stable fast integrators are considered.

We explore next an important result that follow from assuming the previous conjecture to be true. This result states that the set of resonant points is dense in the set $H = \{(h_1, h_2) \in [0, 2/\sqrt{\Lambda_1}] \times \mathbb{R} \mid h_1 \leq h_2\}$. More precisely, this means that for any point $(h_1, h_2) \in H$ and $\varepsilon > 0$, it is possible to find a resonant point $(h_1^{\mathrm{r}}, h_2^{\mathrm{r}})$ such that $|h_1 - h_1^{\mathrm{r}}| + |h_2 - h_2^{\mathrm{r}}| < \varepsilon$.

To prove this result, consider $(h_1, h_2) \in \overset{\circ}{H}$ and $\varepsilon > 0$. Choose p and q such that $(h_1, ph_1/q) \in \overset{\circ}{H}$,

$$\left| \frac{p}{q} h_1 - h_2 \right| < \frac{\varepsilon}{4} \qquad \text{and} \qquad \frac{C}{q\sqrt{\Lambda_1}} < \frac{\varepsilon}{4}. \tag{3.31}$$

It is evident that such a pair (p, q) exists, since p/q can be just adopted to be a rational approximation to h_2/h_1 with q large enough so as to satisfy the second condition in Eq. (3.31). Based on the above conjecture, we have that there exists a resonant point

h_2^{r} such that

$$\left|h_2^r - \frac{p}{q}h_1\right| < \frac{C}{q\sqrt{\Lambda_1}} < \frac{\varepsilon}{4}. \tag{3.32}$$

Together, Eqs. (3.31) and (3.32) imply that $|h_2 - h_2^r| < \varepsilon/2$. Since $h_1^{\mathrm{r}} = h_2^{\mathrm{r}} q/p$ and $q/p \leq 1$, we have from Eq. (3.32) that $|h_1 - h_1^r| < \varepsilon/2$, from where the result follows for any point in $\overset{\circ}{H}$ and hence in H.

We illustrate this result with an example next, in which we arbitrarily selected a set of time steps

$$(h_1, h_2) = (0.0090579193, 0.12698681),$$

and find a pair of resonant time steps nearby. In this case, we chose $p = 85158$ and $q = 6075$ such that $p/q \approx h_2/h_1$. An unstable point over the line $h_2 = ph_1/q$ was found at

$$(h_1^{\mathrm{r}}, h_2^{\mathrm{r}}) \approx (0.0090523798, 0.12690914).$$

At this point we have

$$\mathrm{Tr}(\mathbf{Q}_{\mathrm{AVI}}) \approx -2 - 1.2350353 \times 10^{-16},$$

which is a very weak resonance. By choosing an even larger value for q, and hence p, we could have found an even closer point.

A seemingly unusual consequence of the existence of a dense set of resonant points is that there are instabilities with arbitrarily small time steps (h_1, h_2). As an example, we chose $q = 10{,}000$, $p = 1024\,q + 1$, $\Lambda_1 = \pi^2$, $\Lambda_2 = \pi^2/64$. The first resonant point along the line $h_2 = ph_1/q$ was found at

$$h_1 \approx 9.6902126 \times 10^{-8},\ h_2 \approx 9.9227787 \times 10^{-5}.$$

At this resonant point, the trace is $-2 - 1.2873196 \times 10^{-32}$. The length of the instability interval is approximately 7×10^{-21}. Both features are depicted in Fig. 3.4, which shows the value of the trace of $\mathbf{Q}_{\mathrm{AVI}}$ near its first minimizer. Notice that both h_1 and h_2 are much smaller than the upper bound for stability of the fast integrator, $2/\pi$. In general, the larger the values of p and q, the smaller the values of the first

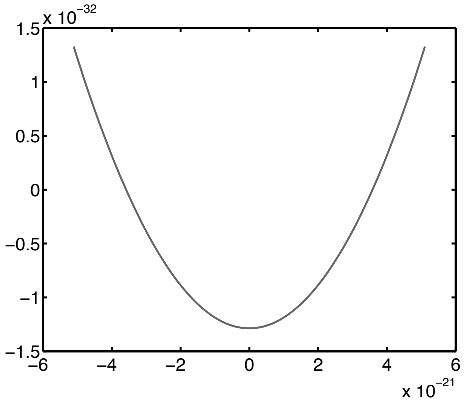


Figure 3.4: Trace of $\mathbf{Q}_{\text{AVI}} + 2$ as a function of $h_2 - h_2^0$, with $h_1 = qh_2/p$ and $h_2^0 \approx 9.9227787 \times 10^{-5}$ being the smallest unstable value. The rest of the parameters for this calculation are: $q = 10{,}000$, $p = 1024\,q + 1$, $\Lambda_1 = \pi^2$, $\Lambda_2 = \pi^2/64$. Negative values on the vertical axis correspond to unstable values of h_2.

resonant set of time steps.

3.3.3 Unstable curves

We next discuss some additional aspects of the numerical experiments. As mentioned, resonances have been found at each local extremum, located at approximately $h_2 = mT_{1/2}(h_1)/q$. However, the vast majority of them are very weak. The largest resonances are near points for which $h_2/T_{1/2}$ is an integer. These resonances are the same type of resonances found in r-RESPA (see result on page 36 for r-RESPA), and are indicated using a circle on Fig. 3.3.

In between the strong resonances near $h_2 = iT_{1/2}$ and $h_2 = (i+1)T_{1/2}$ there are $q-1$ weaker ones. We number the resonances consecutively along the line $h_2 = \frac{p}{q}h_1$ as h_2 grows with an index $m = 1, 2, \ldots$. With this convention, r-RESPA resonances correspond to $m = 0 \bmod(q)$. The next largest resonances were observed to consistently correspond approximately to $m = \pm p \bmod(q)$. More generally, let a, $\lfloor -q/2 \rfloor + 1 \le a \le \lfloor q/2 \rfloor$, be the unique integer such that $m = ap \bmod(q)$ (with p and q having the greatest common denominator equal to 1). The strength of the resonance and the width of the associated resonant interval were consistently observed to *decrease* with $|a|$. In practice, this means that resonances with low values of $|a|$, such as $a = 0$, $a = 1$ and $a = -1$, are the ones which are most likely to be encountered, since the others are very narrow. In Fig. 3.3(a) the square corresponds to $a = 1$ and the diamond corresponds to $a = -1$.

These observations are illustrated in more detail in Figs. 3.5 and 3.6, which show the width of the resonant interval and the amplitude for each resonance, respectively, as a function of the extremum index m. For this example we adopted $q = 1009$ and $p = 2439$. In general, the width of each resonance is highly correlated with the magnitude of the trace. The r-RESPA resonances are shown with circles, and are indeed the dominant ones. The resonances for $a = -3, -2, -1, 1, 2, 3$ are shown with squares and are larger and wider. The remaining resonances are narrower and smaller (with a few exceptions). In this case all extrema were found to be unstable as well, and indeed there are exactly 1008 resonances between the two r-RESPA peaks. However, as emphasized by the number of decades spanned by the logarithmic vertical axis in Fig. 3.6, most resonances are extremely weak. For most of them it would take millions of time steps or more to observe any visible drift in the energy.

The fact that only resonances with relatively low value of $|a|$ are likely to be encountered is nicely displayed by the following numerical calculation. In this case, pairs of time steps are selected such that they are both integer multiples of a given grid spacing h. For this study we adopted $h = 0.0005$, $\Lambda_1 = \pi^2$, $\Lambda_2 = \pi^2/25$, $h \leq h_1 \leq 2/\pi$, and $h_2 \leq 3.5$. If the propagation matrix $\mathbf{Q}_{\mathrm{AVI}}$ for a given pair of time steps (h_1, h_2) has a spectral radius greater than 1, the pair is marked by a dark point and the algorithm is unstable for that choice of time steps. Fig. 3.7 shows the resulting plot over the domain $[h, 2/\pi] \times [h, 3.5]$.

Let us consider first the top plot in Fig. 3.7. The first noteworthy feature is that dark points do not appear everywhere, as would be expected from the fact that resonant pairs form a dense set. Instead, some curves stand out in the midst of a nonuniform cloud of dark points. This is only an artifact of choosing a finite step size, $h = 0.0005$. Large resonant intervals are the only ones likely to be visible on a plot with a finite resolution. As h goes to zero, the figure would be filled with more and more dark dots and the lines would effectively “disappear”.

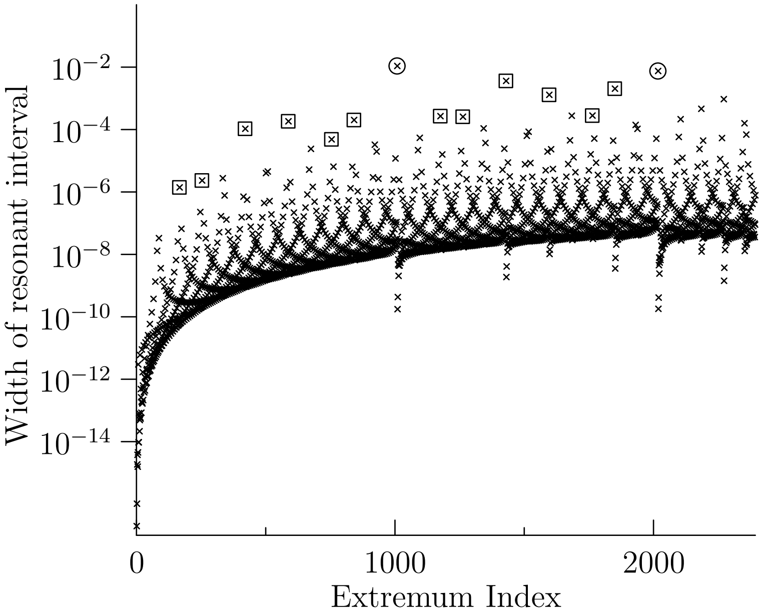


Figure 3.5: Width of the resonant interval as a function of the resonance index for $q = 1009$, $p = 2439$, $\lambda_1 = \pi^2$ and $\lambda_2 = (\pi/8)^2$. The vertical axis is in logarithmic scale. Resonances of the r-RESPA type , in which h_2 is a multiple of $T_{1/2}$, are highlighted with circles. Resonances corresponding to $m = ap \bmod(q)$, with $a = -3, -2, -1, 1, 2, 3$, are highlighted with squares. Extended precision arithmetic was adopted for this calculation to accurately compute the wide span of decades in the vertical axis, including the width of resonances 1 and 2 on the left.

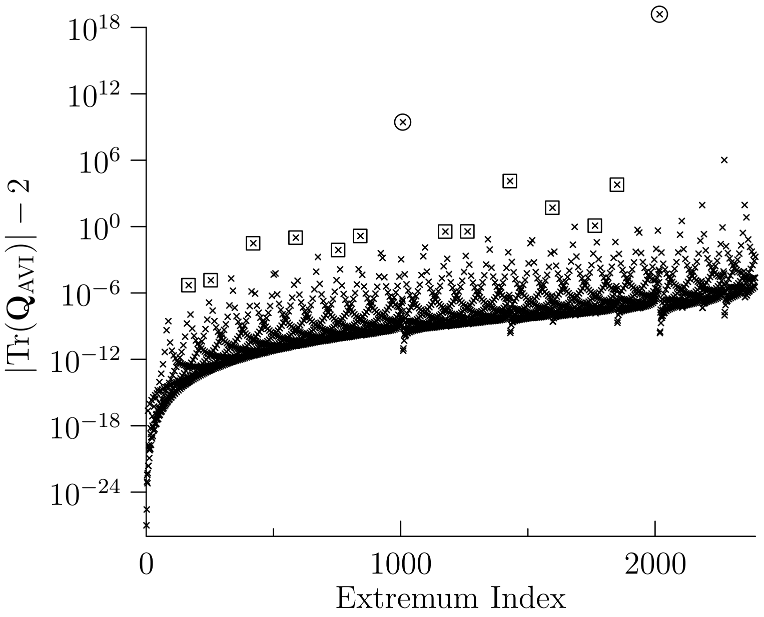


Figure 3.6: Amplitude of each resonance, illustrated as $|\mathrm{Tr}(\mathbf{Q}_{\mathrm{AVI}})| - 2$ on the vertical axis, as a function of the resonance index for the case depicted in Fig. 3.5. The vertical axis is in logarithmic scale. As in Fig. 3.5, resonances of the r-RESPA type are highlighted with circles, while those corresponding to $m = ap \bmod(q)$, with $a = -3, -2, -1, 1, 2, 3$, are highlighted with squares.

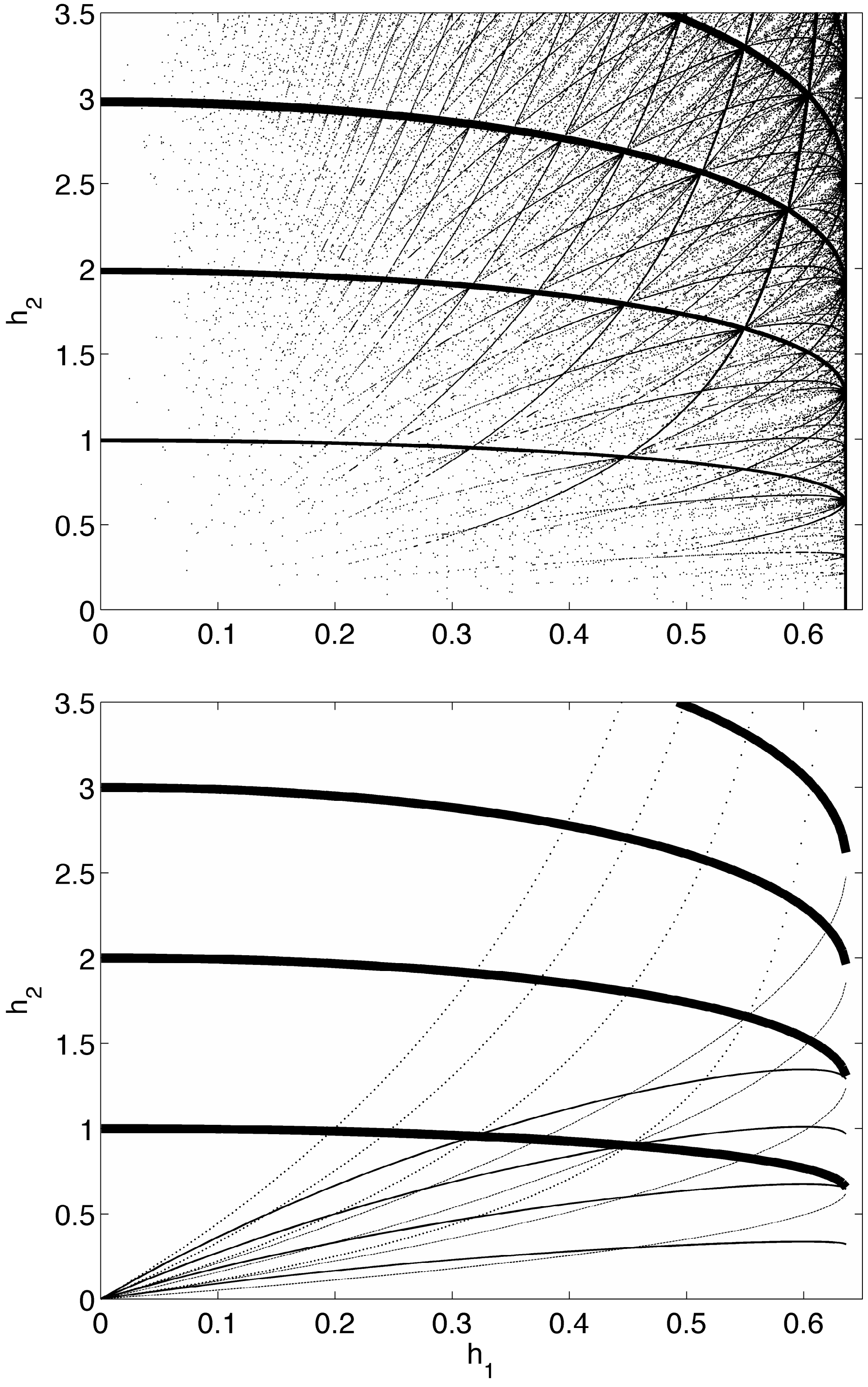


Figure 3.7: AVI stability plot. Top figure: each unstable pair (h_1, h_2) is shown with a dot. Bottom figure: the theoretical prediction given by $b/h_2 = 1/T_{1/2} - a/h_1$ for some pairs of a and b is shown $(a = 0, -1, 1, 2)$. The matching with the numerical results is excellent.

Approximate equations for these curves can be derived based on the previous empirical observations. Each of the thick nearly “horizontal” lines on Fig. 3.7 correspond to a different integer value of $h_2/T_{1/2}$, the r-RESPA resonances. The remaining curves are clearly narrower resonances, which we shall next see that correspond to low values of $|a|$.

Recall that a satisfies that $m = ap \bmod(q)$, from where it follows that $m = ap + bq$ for some integer b. Consequently, if $ph_1 = qh_2$, the resonance condition $qh_2 \approx mT_{1/2}$ is satisfied if and only if

$$\frac{b}{h_2} \approx \frac{1}{T_{1/2}} - \frac{a}{h_1}. \tag{3.33}$$

For given values of a and b this is the equation of a curve in the (h_1, h_2)-plane.

On Fig. 3.7 (bottom row), we plotted the curves with $a = 0$ using thick solid lines (r-RESPA case), $a = 1$ with dotted lines, $a = 2$ with the dashed lines, and $a = -1$ with thin solid lines; b was varied to obtain several curves. The agreement with the location of the resonant points found numerically (Fig. 3.7, top row) is excellent. This comparison highlights the importance of $|a|$ in determining the width of the resonant interval. It would be interesting to obtain an analytical relation between the two that extends the asymptotic results in Section 3.3.

3.4 Why are AVI resonances ubiquitous in molecular dynamics but not in solid dynamics simulations?

Resonances are strong and common in molecular dynamics but seldom appear in solid dynamics simulations. We now use the insight gained in the last two sections with the stability analysis of the one-degree of freedom system to provide an explanation of the startling differences encountered on the performance of AVI in molecular dynamics and solid dynamics simulations. To this end, we performed numerical studies on two simplified systems. The first one resembles a molecular dynamics calculation; weak long-range forces are strongly coupled to local stiff springs. We analyzed the nature of the resonances by applying the results derived in the previous section. The second study considers the analog to a solid dynamics calculation performed with a finite-element discretization, and consists of a mesh of springs with different stiffnesses. Individual time steps for each spring are considered, with smaller time steps assigned to stiffer springs. We will see that in this case resonances are present but they are weak and very narrow, in stark contrast with the first example. This explains in part why such resonances are seldom seen in finite-element calculations.

3.4.1 Molecular dynamics analog

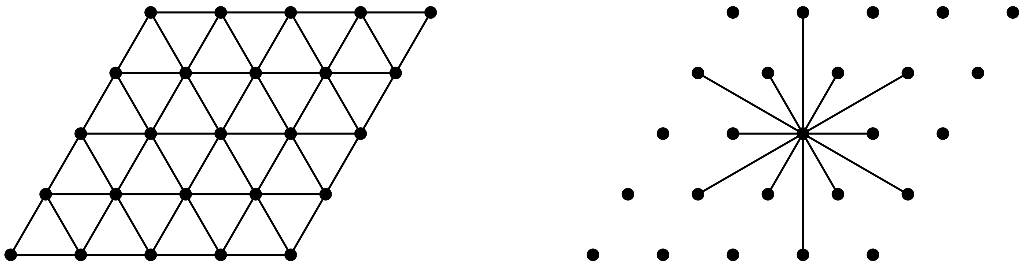
In molecular dynamics, particles are concurrently affected by potentials whose stiffnesses vary greatly. For example the bond potential is very stiff while the electrostatic potential at large distances is very soft. To study the resonant behavior in molecular dynamics we set up an analog using an infinite and periodic 2-D triangular *harmonic* lattice with a unit cell that consists of an $n \times n$ mesh of equal masses. To model the presence of short-range and long-range potentials, stiff springs are used to connect each pair of neighboring masses and a weak gravitational potential connects each mass to its nearest and second nearest neighbors; see Figs. 3.8(a) and 3.8(b) for the unit cell and a sketch of the interactions in the case $n = 4$.

The infinite lattice is formed by locating an identical mass with mass m at every position of the form $(iL, jL\sqrt{3}/2)$ with respect to a pair of Cartesian coordinate axes, for any $(i, j) \in \mathbb{Z}^2$, where L is the lattice parameter. The displacement of the mass at $(iL, jL\sqrt{3}/2)$ is denoted with $(u_{(i,j)}, v_{(i,j)})$, where u and v are the Cartesian components of the displacement vector. Periodic boundary conditions are imposed by restricting the set of possible displacements to those that are periodic with period nL, i.e., $u_{(i,j)} = u_{(i+n,j+n)}$, for all $(i, j) \in \mathbb{Z}^2$, and similarly for v.

Each mass is connected to its six neighboring masses with a spring. The potential energy for each one of these springs is

$$V_{\mathrm{s}}(\ell) = \frac{k}{2}\left(\ell - L\right)^2 , \tag{3.34}$$

where ℓ is the deformed length of the spring. The harmonic lattice corresponds to



(a) Short-range: stiff springs

(b) Long-range: gravitational potential

Figure 3.8: Unit cell of a periodic harmonic lattice for the molecular dynamics analog. Each node in the graph represents a mass, and each edge an interaction between the two nodes at its ends. Short-range interactions are depicted on the left, while long-range ones are indicated on the right.

replacing each one of these springs by its quadratic approximation at $\ell = L$, the equilibrium length of the springs in the lattice in the absence of the gravitational potential. If $(\Delta u, \Delta v)$ indicates the Cartesian components of the difference in displacements between the two ends of the spring, the harmonic approximation to the potential is

$$V_{\mathrm{s}}^{\mathrm{h}}(\Delta u, \Delta v) = \frac{k}{2}(\Delta u \cos\theta + \Delta v \sin\theta)^2, \tag{3.35}$$

where θ is the angle the spring forms with the $(1, 0)$ direction in the undeformed lattice (see Appendix A.2 for details). This is the potential used for each one of the springs in the numerical examples herein, for which we also adopted $n = 4$, $L = 1$, $m = 1$ and $k = 1$.

In the absence of the non-linear gravitational potential, the harmonic lattice possesses a constant symmetric stiffness matrix $\mathbf{K}$ independent of the displacements of the masses. It is then possible to find an orthonormal basis of eigenvectors for $\mathbf{K}$, the eigenmodes of the lattice. For the particular lattices considered here, there exist five groups of eigenmodes with natural frequencies $\omega = \sqrt{6}$, $\sqrt{5}$, $\sqrt{3}$, $\sqrt{2}$, and 1.

Finally, the gravitational potential used to represent the long-range interactions of the masses is given by

$$V_{\mathrm{g}}(r) = -\frac{G}{\sqrt{r^2+\epsilon}}$$

where r is the distance between the two masses, G is the gravitational constant, and ϵ is the softening term. To restrict $V_{\mathrm{g}}(r)$ such that the potential affects only the nearest and second nearest neighbors of each mass $V_{\mathrm{g}}(r)$ is premultiplied by a switching function $S(r)$. Define the modified potential as

$$\tilde{V}_{\mathrm{g}}(r) = \begin{cases} S(r/r_{\mathrm{c}})V_{\mathrm{g}}(r) & \text{if } r \le r_{\mathrm{c}} \\ 0 & \text{if } r > r_{\mathrm{c}} \end{cases}$$

where

$$S(r) = 1 - 10r^3 + 15r^4 - 6r^5$$

and r_{c} is the cutoff radius. The function $S(r)$ is constructed such that $\tilde{V}_{\mathrm{g}}(0) = V_{\mathrm{g}}(0)$ and $\tilde{V}_{\mathrm{g}}(r)$ is a C^2 smooth function. Nearest neighbors in the lattice are separated by the lattice parameter L, while second nearest neighbors by $\sqrt{3}L$, so by adopting $\sqrt{3}L < r_{\mathrm{c}} \le 2L$ each mass in the periodic harmonic lattice is only affected by the long-range potential through interactions with its nearest and second nearest neighbors. Fig. 3.9 compares these two versions of the gravitational potential. For our simulations we adopted $\tilde{V}_{\mathrm{g}}(r)$ as the weak long-range potential and set $G = 0.01$, $\epsilon = 1$, and $r_{\mathrm{c}} = 1.85L$.

In the absence of the gravitational potential, each one of the eigenmodes of the lattice is itself an independent harmonic oscillator that does not interact with the remaining eigenmodes. The introduction of the long-range potential breaks this isolation, and induces a weak interaction among all eigenmodes. This is sketched in Fig. 3.10.

The total energy of each mass in the lattice is defined as the sum of its kinetic energy plus its potential energy contributions. The latter is formed by adding up one-half of the potential energy of each harmonic spring and each gravitational interaction attached to the mass. The total energy in the unit cell of the lattice is obtained as the sum of the total energy of each mass in the cell. For later use, it is also useful to

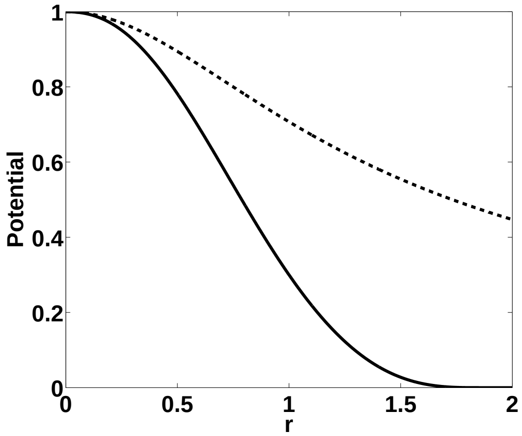


Figure 3.9: Comparison of the two versions of the gravitational potential. The dotted line represents the gravitational potential $V_\mathrm{g}(r) = -G/\sqrt{r^2+\epsilon}$ with $G = 1$ and $\epsilon = 1$. The solid line is the modified potential $\tilde{V}_\mathrm{g}(r)$ which equals $-S(r/r_\mathrm{c})G/\sqrt{r^2+\epsilon}$ for $0 \le r \le r_\mathrm{c}$ (where $S(r) = 1 - 10r^3 + 15r^4 - 6r^5$) and zero for $r > r_\mathrm{c}$. Here r_c is taken to be 1.85.

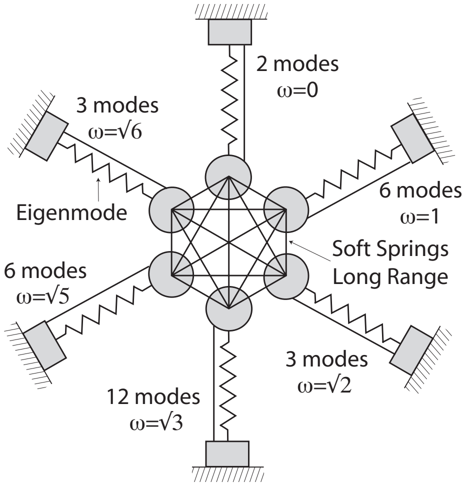


Figure 3.10: Schematic interpretation of the lattice with a weak gravitational potential. In the absence of the gravitational potential each eigenmode in the lattice is an independent harmonic oscillator, depicted here as each one of the six wiggly springs. The gravitational potential breaks this isolation by connecting together the masses of each one of the eigenmodes via weak springs, depicted here with straight segments.

specify a potential energy for each eigenmode in the unit cell, taken equal to $u_\omega^\mathsf{T}\mathbf{K}u_\omega/2$, where u_ω is the displacement along the eigenmode with frequency ω.

For the numerical experiments we chose only two different time steps: a small time step h_1 for the springs, the stiff potentials; and a larger one h_2 for the gravitational potential, which is soft. In fact we chose $h_1 = 0.001$ so that the integration of each one of the eigenmodes of the harmonic lattice is nearly exact, because $h_1\omega \ll 1$ for any of the frequencies listed earlier. Since the gravitational potential is weak, its effect over each one of the eigenmodes is similar to that of the soft potential in the analysis in Sections 3.2 and 3.3, and hence we expect that if h_2 is close to an integer multiple of the half-period of any of the eigenmodes, a numerical resonance should be observed.

To search for resonances the time step h_2 for the gravitational potential was varied from 1 to 4 in increments of 0.005. Each integration was carried out until a maximum time of 500 was reached. At the end of each simulation the total energy of the unit cell lattice was recorded. The energy growth was computed as the difference between the total energy at the beginning and at the end of the simulation. These results are shown in Fig. 3.11, which also shows with solid dots the values of h_2 that correspond to integer multiples of the half-period for each one of the five different values of ω in the simulation. The close location of the spikes in the figure to the solid dots evidences that the expected numerical resonances are in fact occurring, triggered by the interaction of the gravitational potential with each one of the eigenmodes.

To further examine this resonant behavior, we chose the time step h_2 to be close to the half-period of the $\omega = \sqrt{6}$ and $\omega = \sqrt{5}$ eigenmodes. Their half-periods are $T_{1/2} = \pi/\sqrt{6} \approx 1.283$ and $T_{1/2} = \pi/\sqrt{5} \approx 1.405$, respectively. The total potential energy of the $\omega = \sqrt{6}$ eigenmodes for $h_2 = 1.285$ and the $\omega = \sqrt{5}$ eigenmodes for $h_2 = 1.41$ are shown in Figs. 3.12 and 3.13, along with the energy of the other eigenmodes. In both cases we observe an exponential-in-time growth of the total energy of the resonant eigenmodes, with a nearly unaffected evolution for the remaining ones. Only very late in the simulation does Fig. 3.13(b) display the beginning of an exponential-in-time growth of the energy of one of the remaining eigenmodes, as a result of the weak coupling between them and the already large amplitude of the $\omega = \sqrt{5}$ eigenmodes.

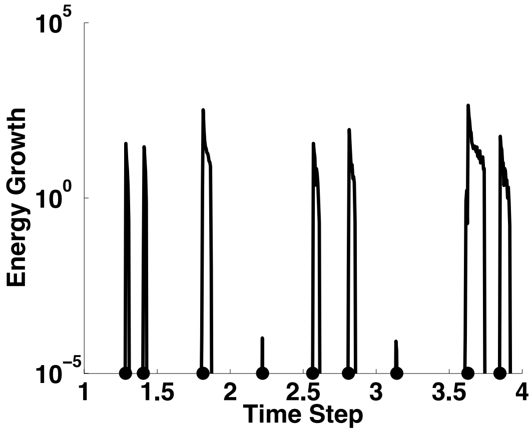
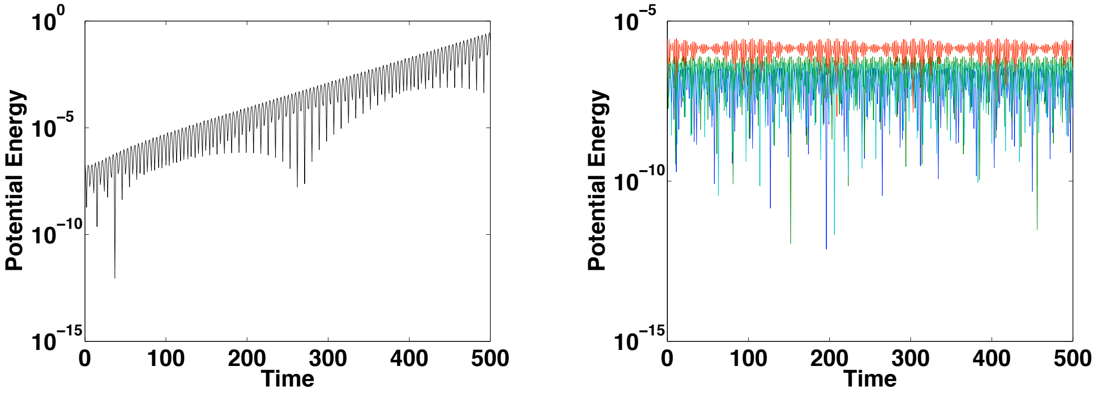


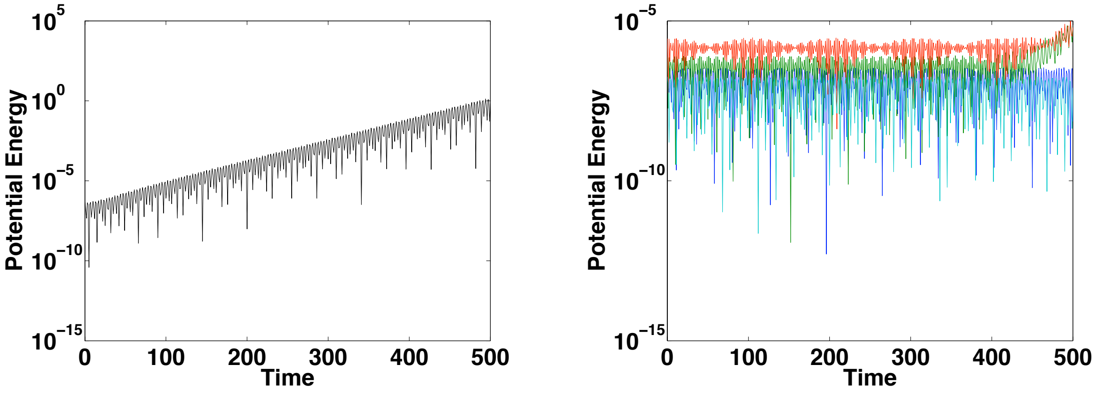
Figure 3.11: Energy growth of the unit cell of the lattice due to numerical instabilities, as a function of the time step h_2 used to integrate the long-range gravitational potential. The energy growth at each time step value was computed as the difference between the energy at the beginning and at the end of each simulation. The solid dots represent the expected resonant time steps according to Section 3.2.



(a) $\omega = \sqrt{6}$

(b) Other eigenmodes. Each color depicts a group of eigenmodes with the same natural frequency.

Figure 3.12: Evolution of the potential energy for each group of eigenmodes for $h_2 = 1.285$, the time step of the weak long-range gravitational potential. It is seen that the eigenmodes whose half-period is $\pi/\sqrt{6} \approx 1.283$ are resonant with the long-range interaction, as expected from Section 3.2.



(a) $\omega = \sqrt{5}$

(b) Other eigenmodes. Each color depicts a group of eigenmodes with the same natural frequency.

Figure 3.13: Evolution of the potential energy for each group of eigenmodes for $h_2 = 1.41$, the time step of the weak long-range gravitational potential. It is seen that the eigenmodes whose half-period is $\pi/\sqrt{5} \approx 1.405$ are resonant with the long-range interaction, as expected from Section 3.2. Only by the end of the simulation does another group of eigenmodes display exponential-in-time growth of the potential energy, as a result of the weak coupling with the by-then-large amplitude oscillations of eigenmodes corresponding to $\omega = \sqrt{5}$ through the long-range potential.

For a molecular dynamics simulation with a large number of atoms, integer multiples of the half-periods of the eigenmodes (also called the normal modes) are essentially closely packed over the real line. Therefore, in practice, it is virtually impossible to choose a value for h_2 that does not resonate with at least one of the eigenmodes, as illustrated with this example. Remarkably, the results on the very simple one-degree of freedom system in Sections 3.2 and 3.3 provide a clear explanation of the behavior observed in this example, and can be extrapolated to infer some aspects of the behavior for more complex systems. Fundamentally, what makes it possible is precisely the apparent lack of interaction among different modes of the lattice at the early stages of the resonant behavior.

3.4.2 Solid dynamics analog

The second set of studies focus on a finite-element-like simulation, such as those used in solid mechanics for linear elastic problems. These cases usually lack a long-range force and are characterized by the existence of a range of element stiffness values, arising from the presence of different material properties and element sizes. Moreover, in many situations element stiffness values vary smoothly throughout the domain.

The potential energy for an elastic solid can be written as a sum of elemental contributions. An AVI discretization then naturally follows by assigning a possibly-different time step to each one of the elements, see [58]. The time step of each element is inversely proportional to its element stiffness value, and hence softer elements are allocated larger time steps.

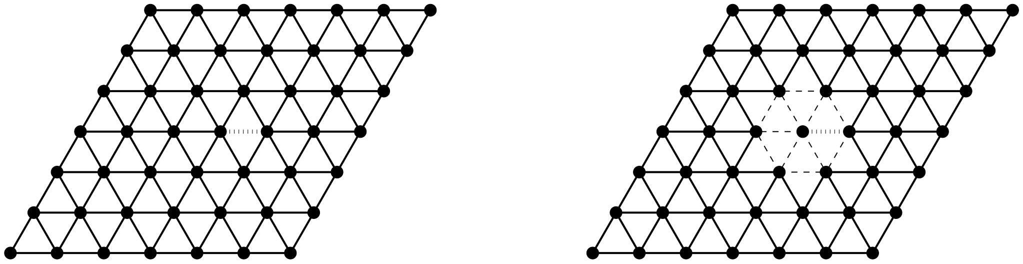
If adaptive mesh refinement strategies are adopted, then the range of elemental stiffness values could possibly be very wide. In fact, some elements in the mesh are often integrated with time steps that are near an integer multiple of the half-period of a resonant mode of the structure. Why is then that the expected resonances are generally not encountered in practice, as evidenced in the simulations in [57, 58, 59]? The answer is that the resonances are still present, as the foregoing analysis demonstrates. However, their amplitude and width are so small that they are hardly noticed in practice. The key difference with the molecular dynamics case in Section 3.4.1 lies in the absence of the long-range force. We shall illustrate these ideas with an example next.

We consider herein the same harmonic lattice adopted for the example in Section 3.4.1, in this case with no long-range gravitational potential and with a larger unit cell. The potential energy for each one of the springs connecting two neighboring masses is again determined by the harmonic approximation in Eq. (3.35), and displacements are restricted to be periodic, as specified earlier. When all springs are identical, the stiffness matrix of this lattice is identical to that obtained by identifying each triangle in the lattice as a piecewise linear finite-element made of an isotropic linear elastic material with Lame constants $\lambda = \mu = \sqrt{3}k/4$.

To represent the spatially varying stiffness values, we consider two different configurations of the springs: 1) only one spring in the unit cell is soft, while all the remaining ones are stiff, see Fig. 3.14(a); and 2) a *core* of several soft springs, while the rest of the springs in the unit cell are stiff, as shown in Fig. 3.14(b). In both instances, however, all springs but one are integrated with a small time step h_1, while one of the soft springs, the same in both cases, is integrated with a longer time step h_2, see Fig. 3.14. For both types of lattices, the relevant eigenmodes to study the resonant behavior are those of the unit cell without the only soft spring integrated with a longer time step. We shall henceforth refer to these as the eigenmodes.

For all the forthcoming numerical examples we have adopted the values of $n = 7$, $L = 1$ and $k = 1$ for the stiff springs and $k = 0.33$ for the soft ones. We have purposely decided to avoid making the latter much smaller than the former, so as to highlight other types of phenomena that may also occur, as we shall next see. As in Section 3.4.1, we set $h_1 = 0.001$ so that it is much smaller than the period of any natural frequency in the lattice. The time step h_2 was varied from 1.3 to 1.6 in increments of 0.005. The total energy of the system was recorded at the end of a simulation with a total simulation time equal to 500.

Fig. 3.15 shows the total value of the energy at the end of each simulation as a function of h_2 for the first case, i.e., when only one soft spring is considered. The dots on the horizontal axis represent the half-period of the eigenmodes that fall within



(a) One soft spring

(b) Core of soft springs

Figure 3.14: Solid mechanics analogs using harmonic lattices. The soft spring integrated with a large time step h_2 is drawn with a dotted line. In the case depicted on the right the rest of the soft springs, which are integrated with a small time step h_1, are drawn with a dashed line. Stiff springs are drawn with a solid line.

this time step range and are coupled to the soft spring (only three eigenmodes). As before, when the time step h_2 is near the half-period of one of the eigenmodes, resonant behavior was observed. However, in this case the resonance plot is not as simple as that in Section 3.4.1. In addition to the resonance instabilities that correspond to each one of the natural frequencies of the system, we observe a number of other peaks between the dots on the horizontal axis. This more complex profile results from a coupling between modes whose frequencies are close to one another given that the stiffness of the soft springs, 0.33, is not much smaller than that of the stiff ones, 1. These instabilities can be understood by analyzing a system of two harmonic oscillators joined by a third spring, in which the former are integrated with a smaller time step than the latter. We shall not expand on this observation, but just note that other types of resonances can be observed in addition to those arising from the excitation of a single eigenmode.

We now turn to the case where we have a core of soft springs, but in which only one of them is integrated with a long time step h_2. The total energy at the end of the simulation as a function of h_2 is shown in Fig. 3.16. The marks on the horizontal axis indicate the half-periods of those eigenmodes that fall within this time step range and

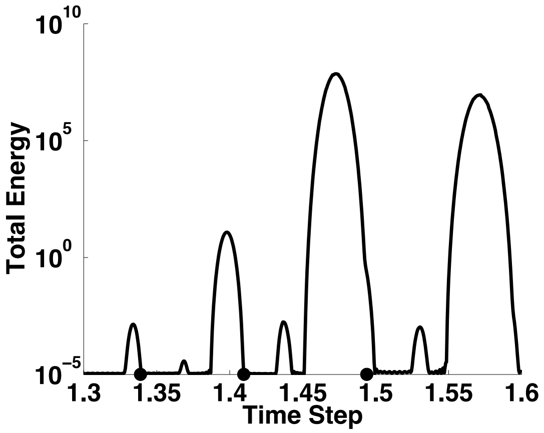


Figure 3.15: Total energy of the harmonic lattice as a function of the time step adopted for the integration of the single soft spring in the lattice. The dots on the horizontal axis indicate integer multiples of half-periods of the eigenmodes in this time interval. Only those eigenmodes that are coupled to the soft spring are shown. Two types of resonances are observed, those that excite eigenmodes with the same natural frequency, recognized by the dots within them, and those that excite combinations of eigenmodes, as detailed in the text.

have some non-negligible coupling with the soft spring integrated with a long time step. We observe that only the energy peak located near $h_2 = 1.56$ corresponds to one of the natural frequencies of the lattice (dot), with the other five peaks resulting from coupling between two or three eigenmodes. The marks at the energy peaks identify which modes are involved in the coupling that produces the observed instability. The eigenmode with half-period near 1.56 is involved in all five coupling events. For example the peak located at 1.45 is the product of coupling between the eigenmode with a half-period near 1.35 (triangle) and the eigenmode with a half-period near 1.56 (dot). In addition notice that the vertical energy scale is much shorter than that in Fig. 3.15. Resonances are still present, but they have been severely tamed and are much narrower than those in Fig. 3.15 when the soft core was absent.

To explain this result, it is convenient to take a look at the stiffness matrix of the entire lattice $\mathbf{K}$. We can decompose $\mathbf{K}$ as $\mathbf{K} = \mathbf{K_1} + \mathbf{K_2}$ where $\mathbf{K_2}$ contains the one soft spring being integrated with the long time step and $\mathbf{K_1}$ contains all of the other springs (containing in fact both stiff and soft springs). The equation of motion

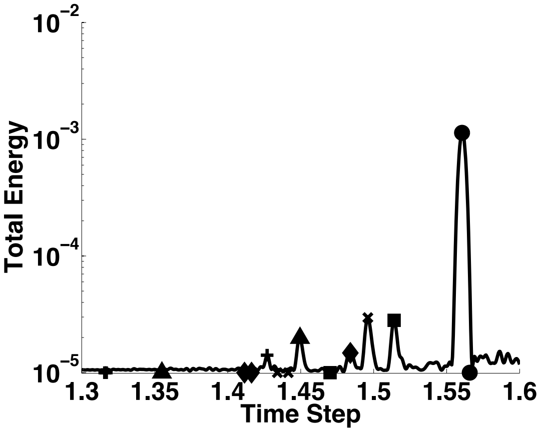


Figure 3.16: Total energy of the harmonic lattice with a core of soft springs, in which only one of these springs is integrated with a longer time step. This time step is indicated on the horizontal axis. The marks on the horizontal axis indicate the half-periods of the eigenmodes that have a non-negligible coupling with the spring with the long time step. Observe that only the energy peak denoted by the dot occurs at one of the natural frequencies of the lattice. The other peaks result from a coupling between modes with the same mark and the mode marked by a dot. For example the energy peak at 1.45 is due to a coupling between the mode with half-period 1.35 (triangle) and the mode with half-period 1.56 (dot). Notice that, in comparison with Fig. 3.15, the vertical axis has a drastically shorter energy scale, reflecting the fact that resonances in this case are much weaker and narrower than when only one soft spring is present in the entire lattice.

is given by

$$\ddot{x} + (\mathbf{K_1} + \mathbf{K_2})\, x = 0.$$

Now we can perform an eigenvector transformation for $\mathbf{K_1}$ which gives

$$\ddot{y} + (\mathbf{\Lambda} + \mathbf{V}^{\mathrm{T}}\mathbf{K_2}\mathbf{V})\, y = 0$$

where $\mathbf{K_1} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{\mathrm{T}}$ and $y = \mathbf{V}^{\mathrm{T}}x$. This equation shows that, similar to the molecular dynamics case, there is a coupling between the soft spring with a long time step and the stiff modes. However this coupling is given by matrix $\mathbf{V}^{\mathrm{T}}\mathbf{K_2}\mathbf{V}$. Because of the presence of the soft core, the entries in $\mathbf{V}^{\mathrm{T}}\mathbf{K_2}\mathbf{V}$ corresponding to stiff modes are very small, since as we shall see, stiff eigenmodes decay rapidly once they enter the region with soft springs. It then follows that the very small coupling terms in $\mathbf{V}^{\mathrm{T}}\mathbf{K_2}\mathbf{V}$ for the stiff modes effectively makes the soft spring integrated with time step h_2 even softer. As can be seen from Eqs. (3.7) and (3.8), the resulting resonances are still present, but they will be narrow and weak. Therefore, resonances are seldom observed in these types of simulations.

We showcase next the exponential decay of the high-frequency eigenvectors in the soft region through a one-dimensional example. Consider a system of 20 springs made of 10 springs with stiffness $k_1 = 8$ attached to 10 other springs with stiffness $k_2 = 1$, and identical masses between any two consecutive springs. The eigenvectors must satisfy the following equations:

$$k^{(i-1)}(x_i - x_{i-1}) - k^{(i)}(x_{i+1} - x_i) = \lambda x_i, \quad i = 2, \ldots, 20$$

where x_i indicates the position of mass i and $k^{(i)}$ is the stiffness of spring i which connects masses i and $i+1$. A high-frequency eigenvector has $\lambda \gg k_2$. To estimate its decay in the region with soft springs, we assume that $|x_{i+1}| \ll |x_i|$ for $i \geq 12$. Then in this region

$$x_i \approx \frac{k_2}{2k_2 - \lambda}\, x_{i-1}.$$

This corresponds to a geometric decay with rate $\approx -k_2/\lambda$. Fig. 3.17 plots the first three stiffest eigenvectors. The circles correspond to the predicted geometric decay.

We see that the prediction is quite accurate.

This example demonstrates that stiff eigenvectors decay exponentially fast when they enter a soft region. A similar phenomenon should be valid in two-dimensional and three-dimensional meshes as well. In a mechanical problem then, soft regions of the mesh are only “locally” coupled to stiff ones. A soft element can still induce resonances in nearby stiff regions, but the strength of their coupling is exponentially small with the distance between them.

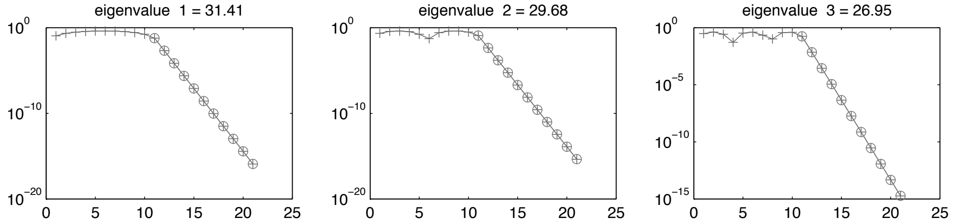


Figure 3.17: Exponential decay of eigenvectors over core of soft springs. The circles are the analytical approximation of the decay. The agreement between the two is rather good.

3.5 Summary

We studied the stability of AVI integrators and showed that results derived for the synchronous r-RESPA family of integrators can be generalized to asynchronous integrators such as AVI. We provided sufficient conditions for stability. We postulated that for a system of two 1-D springs with different stiffnesses an instability is observed when the synchronization time $ph_1 = qh_2$ is near a multiple of the half-period of the stiff potential. We motivated a conjecture that implies that the set of resonant time steps is dense, which was verified using systematic numerical tests. It also follows from the conjecture that arbitrarily small unstable time steps exist and, indeed, very small unstable time steps were found numerically. Most of these resonances were found to be very weak, and of little importance in practice. We characterized the strongest resonances through a family of curves parametrized by two integers, which reproduced the numerical results very well.

In addition, we examined the resonance behavior of AVI in molecular dynamics and solid mechanics. The main result is that resonances are easily observed in molecular dynamics due to the strong coupling between stiff modes and soft long-range forces (e.g. electrostatic forces). On the other hand, resonant behavior is rarely seen in solid mechanics because the smooth gradient of element stiffnesses leads to a very weak coupling and hence very narrow bands of resonant time steps.

Finally, the analyses herein are only concerned with the linear stability of the method. Other important resonances may be induced by the non-linearities in the system. Some results in this direction in the vicinity of stable equilibrium points are presented in [82]. Additionally, we note that classical techniques for enhancing the stability of multiple time stepping schemes such as MOLLY [51], the isokinetic Nosé-Hoover chain RESPA [69], and the two-force method [40, 41] may be applicable to AVI, and may lead to improved robustness for the method. The two-force method is very promising since it removes all instabilities in the integrator. However it currently has two drawbacks: it can be computationally expensive and it does not extend to three or more time steps without exponentially increasing its computational cost. In the next chapter we will discuss how another technique for quelling resonances, Langevin equations [3, 4, 77], can be extended to the AVI setting.

Chapter 4

A Stochastic Variational Integrator

The formulation of AVI presented thus far is only applicable to Hamiltonian systems. However there are situations where it is desirable to explore non-Hamiltonian dynamics. For instance, in practice the size of the time steps used in molecular dynamics simulations is often limited by the presence of resonances. One approach for mitigating the undesirable effect of these resonances is to introduce numerical damping through Langevin dynamics. In addition to quelling resonances Langevin dynamics also functions as a stochastic thermostat and models the influence of explicit water molecules on solvated proteins. Since the dynamics are no longer Hamiltonian, the standard AVI cannot be applied directly to these simulations. In this chapter we describe how a stochastic version of AVI can be derived from the variational framework that is applicable for Langevin dynamics. In addition we show that, depending on how the action is approximated, different stochastic variational integrators (SVI) can be obtained. To illustrate their functionality, we conclude by applying one of the proposed integrators to an MD simulation of a simple peptide.

4.1 Continuous Variational Principle

We begin by extending the variational framework to Langevin dynamics in the continuous setting. Define the stochastic action [add citation] of the system to be

$$S^{\mathrm{s}}[q(\cdot)] = \int_0^T e^{\gamma t}\left(\sum_{a=1}^{N}\frac{1}{2}m_a\left|\left|\dot{q}_a\right|\right|^2 - V(q)\right)dt + \int_0^T e^{\gamma t}\sum_{a=1}^{N}\sum_{\alpha=1}^{d}\sigma_a q_{a\alpha}\circ dW_{a\alpha}(t)$$

where γ is the friction coefficient, $\sigma_a^2 = 2\gamma m_a k_B T$, d is the number of dimensions for the vector q_a, and $W_a(t)$ is a vector of standard white noise, i.e.

$$\langle W_a(t)\rangle = \vec{0}, \quad \langle W_a(t)W_a(t')^T\rangle = \delta(t-t')\mathbf{I}_{d\times d}.$$

Note that this action differs from the one defined in Sec. 2.1 in two respects. First we have introduced an integrating factor $\exp(\gamma t)$ in order to reproduce the friction term in Langevin dynamics. Second a stochastic term has been added to account for the random force. Here the stochastic integral is defined in the Stratonovich sense instead of Ito because the chain rule of ordinary calculus holds for the former but not the latter. This will be helpful as we derive the Euler-Lagrange equations using variational calculus. Applying the variational principle to this action and assuming that the trajectory $q(t)$ is a stationary point gives

$$0 = \delta S^{\mathrm{s}} = \int_0^T e^{\gamma t}\sum_{a=1}^{N}\left(m_a\dot{q}_a\delta\dot{q}_a - \frac{\partial V(q)}{\partial q_a}\delta q_a\right)dt + \int_0^T e^{\gamma t}\sum_{a=1}^{N}\sigma_a\delta q_a\circ dW_a(t).$$

Using integration by parts and applying the boundary conditions $\delta q_a(0) = 0$ and $\delta q_a(T) = 0$ we have

$$
\begin{aligned}
0 &= \left[e^{\gamma t} \sum_{a=1}^{N} m_a \dot{q}_a \delta q_a \right]_{t=0}^{t=T} - \int_0^T \sum_{a=1}^{N} \left(m_a \frac{d}{dt} \left(\dot{q}_a e^{\gamma t} \right) \delta q_a + e^{\gamma t} \frac{\partial V(q)}{\partial q_a} \delta q_a \right) dt \\
&\quad + \int_0^T e^{\gamma t} \sum_{a=1}^{N} \sigma_a \delta q_a \circ dW_a(t) \\
&= - \int_0^T \sum_{a=1}^{N} \left(m_a \left(\ddot{q}_a + \gamma \dot{q}_a \right) + \frac{\partial V(q)}{\partial q_a} \right) \delta q_a e^{\gamma t} dt + \int_0^T e^{\gamma t} \sum_{a=1}^{N} \sigma_a \delta q_a \circ dW_a(t) \\
&= \int_0^T e^{\gamma t} \sum_{a=1}^{N} \delta q_a \left[\left(-m_a \ddot{q}_a - \gamma m_a \dot{q}_a - \frac{\partial V(q)}{\partial q_a} \right) dt + \sigma_a \circ dW_a(t) \right].
\end{aligned}
$$

Since the last equality must hold for all variations $\delta q_a(t)$ the following stochastic differential equation (SDE) needs to be satisfied for each particle:

$$
dp_a = \left(-\frac{\partial V(q)}{\partial q_a} - \gamma p_a \right) dt + \sigma_a dW_a(t)
$$

where $p_a = m_a \dot{q}_a$ is the momentum of coordinate q_a. Note that the Stratonovich integral has been replaced with the Ito integral since the two are identical when the integrand is independent of time. Hence the Euler-Lagrange equations obtained from the proposed stochastic action are exactly the Langevin equations.

4.2 Discrete Variational Principle

Similar to the procedure adopted in Sec. 2.2 stochastic variational integrators for Langevin dynamics can be constructed by mimicking the variational structure in the discrete setting. The idea once again is to approximate the stochastic action over a discrete trajectory and use a discrete variational principle to obtain the integrator. More precisely, the time interval $[0, T]$ is partitioned into a sequence of times $\{t^j\} = \{t^0 = 0, \ldots, t^M = T\}$ with a time step h, and a discrete trajectory on this partition is a sequence of positions $\{q^j\} = \{q^0, \ldots, q^M\}$. To derive an integrator we need to construct a stochastic discrete Lagrangian that approximates the stochastic action over one time step h. This involves approximating the two integrals in the stochastic action. For the deterministic integral, a suitable discrete approximation is given by

$$\begin{aligned}&\int_0^T e^{\gamma t}\left(\sum_{a=1}^{N}\frac{1}{2}m_a\left\|\dot{q}_a\right\|^2 - V(q)\right)dt\\ &\approx \sum_{j=0}^{M-1} e^{\gamma t^{j+1/2}} h\left(\sum_{a=1}^{N}\frac{1}{2}m_a\left\|\frac{q_a^{j+1}-q_a^j}{h}\right\|^2 - \frac{V(q^j)+V(q^{j+1})}{2}\right).\end{aligned}$$

Here we take the same discretization as VV for the kinetic and potential energy terms and simply evaluate the exponential integrating factor at the midpoint of the time interval (t^j, t^{j+1}). The stochastic integral can be approximated as follows:

$$\begin{aligned}&\int_0^T e^{\gamma t}\sum_{a=1}^{N}\sum_{\alpha=1}^{d}\sigma_a q_{a\alpha}\circ dW_{a\alpha}(t)\\ &=\sum_{j=0}^{M-1}\int_{t^j}^{t^{j+1}} e^{\gamma t}\sum_{a=1}^{N}\sum_{\alpha=1}^{d}\sigma_a q_{a\alpha}\circ dW_{a\alpha}(t)\\ &\approx\sum_{j=0}^{M-1}\sum_{a=1}^{N}\sum_{\alpha=1}^{d}\left[q_{a\alpha}^j\sigma_a\int_{t^j}^{t^{j+1/2}} e^{\gamma t}\circ dW_{a\alpha}(t) + q_{a\alpha}^{j+1}\sigma_a\int_{t^{j+1/2}}^{t^{j+1}} e^{\gamma t}\circ dW_{a\alpha}(t)\right].\end{aligned}$$

Here we have taken $q(t)$ to be constant and equal to q^j when t is in the time interval $(t^{j-1/2}, t^{j+1/2})$. These two stochastic integrals can be evaluated exactly by using a result from stochastic calculus [add citation]

$$\sigma_a e^{-\gamma h/2}\int_0^{h/2} e^{\gamma s}\circ dW_a(s) = \sigma_a e^{-\gamma h/2}\int_0^{h/2} e^{\gamma s} dW_a(s) = N\left(0, m_a k_B T\left(1-e^{-\gamma h}\right)\right).$$

The first equality, which states that the Stratonovich and Ito integrals are equal, is a consequence of having an integrand that is a deterministic function of time. In this instance the drift correction term that arises from converting Stratonovich to Ito is identically zero. Applying this result to the approximation of the stochastic integral in the stochastic action gives

$$\begin{aligned}
&\int_0^T e^{\gamma t} \sum_{a=1}^{N} \vec{1} \cdot (\sigma_a q_a \circ dW_a(t)) \\
&\approx \sum_{j=0}^{M-1} \sum_{a=1}^{N} \sum_{\alpha=1}^{d} \left[q^j_{a\alpha} \sigma_a e^{\gamma t^j} \int_0^{h/2} e^{\gamma s} \circ dW_{a\alpha}(s) + q^{j+1}_{a\alpha} e^{\gamma t^{j+1/2}} \sigma_a \int_0^{h/2} e^{\gamma s} \circ dW_{a\alpha}(s) \right] \\
&= \sum_{j=0}^{M-1} \sum_{a=1}^{N} \sum_{\alpha=1}^{d} \left(q^j_{a\alpha} e^{\gamma t^{j+1/2}} \xi^{j+1/2}_{a\alpha} + q^{j+1}_{a\alpha} e^{\gamma t^{j+1}} \xi^{j+1}_{a\alpha} \right)
\end{aligned}$$

where ξ^j_a is a normal random vector with mean $\vec{0}$ and covariance matrix given by $\left(m_a k_B T \left(1 - e^{-\gamma h}\right)\right) \delta_{lm}$. For every time step h the random vector needs to be sampled twice.

Therefore an approximation of the stochastic action is given by the following action sum:

$$S^{\mathrm{s}}_{\mathrm{d}}[\{q^j\}] = \sum_{j=0}^{M-1} L^{\mathrm{s}}_{\mathrm{d}}(q^j, q^{j+1}, j)$$

where the stochastic discrete Lagrangian $L^{\mathrm{s}}_{\mathrm{d}}(q, \tilde{q}, j)$ takes the form

$$\begin{aligned}
L^{\mathrm{s}}_{\mathrm{d}}(q, \tilde{q}, j) = e^{\gamma t^{j+1/2}} h \left(\sum_{a=1}^{N} \frac{1}{2} m_a \left\| \frac{\tilde{q}_a - q_a}{h} \right\|^2 - \frac{V(q) + V(\tilde{q})}{2} \right) \quad (4.1) \\
+ \sum_{a=1}^{N} \sum_{\alpha=1}^{d} \left(q_{a\alpha} e^{\gamma t^{j+1/2}} \xi^{j+1/2}_{a\alpha} + \tilde{q}_{a\alpha} e^{\gamma t^{j+1}} \xi^{j+1}_{a\alpha} \right). \quad (4.2)
\end{aligned}$$

The integrator follows from the discrete Euler-Lagrange equations for this discretization:

$$\frac{\partial}{\partial q} L^{\mathrm{s}}_{\mathrm{d}}(q^j, q^{j+1}, j) + \frac{\partial}{\partial \tilde{q}} L^{\mathrm{s}}_{\mathrm{d}}(q^{j-1}, q^j, j-1) = 0,$$

for $j = 1, \ldots, M-1$, where $\partial L^{\mathrm{s}}_{\mathrm{d}}(\cdot, \cdot)/\partial q$ and $\partial L^{\mathrm{s}}_{\mathrm{d}}(\cdot, \cdot)/\partial \tilde{q}$ indicate the partial derivatives of $L^{\mathrm{s}}_{\mathrm{d}}$ with respect to its first and second arguments, respectively. The discrete momenta $\{p^j\}$ are generally defined, via a discrete Legendre transform, to be

$$p_a^j e^{\gamma t^j} = \frac{\partial}{\partial \tilde{q}_a} L_{\mathrm{d}}^{\mathrm{s}}(q^{j-1}, q^j, j-1) = -\frac{\partial}{\partial q_a} L_{\mathrm{d}}^{\mathrm{s}}(q^j, q^{j+1}, j), \tag{4.3}$$

where the second equality follows from the discrete Euler-Lagrange equations. Comparing this expression to (2.2) note that the momenta are scaled by the integrating factor. This is a consequence of introducing the integrating factor $\exp(\gamma t)$ into the definition of the stochastic action. Substituting the stochastic discrete Lagrangian (4.1) into Eq. (4.3) gives:

$$\begin{aligned} p_a^j e^{\gamma t^j} &= e^{\gamma t^{j-1/2}} m_a \left(\frac{q_a^j - q_a^{j-1}}{h} \right) - e^{\gamma t^{j-1/2}} \frac{h}{2} \frac{\partial V}{\partial q_a}(q_a^j) + e^{\gamma t^j} \xi_a^j \\ p_a^j e^{\gamma t^j} &= e^{\gamma t^{j+1/2}} m_a \left(\frac{q_a^{j+1} - q_a^j}{h} \right) + e^{\gamma t^{j+1/2}} \frac{h}{2} \frac{\partial V}{\partial q_a}(q_a^j) - e^{\gamma t^{j+1/2}} \xi_a^{j+1/2}. \end{aligned}$$

The second momentum equation can be rearranged to give

$$p_a^{j+1/2} = e^{-\gamma h/2} p_a^j - \frac{h}{2} \frac{\partial V}{\partial q_a}(q_a^j) + \xi_a^{j+1/2}$$

while the first equation, replacing j with $j+1$, results in

$$p_a^{j+1} = e^{-\gamma h/2} \left(p_a^{j+1/2} - \frac{h}{2} \frac{\partial V}{\partial q_a}(q_a^{j+1}) \right) + \xi_a^{j+1}.$$

Hence the numerical integrator arising from the stochastic discrete Lagrangian (4.1) is

$$\begin{aligned} p_a^* &= e^{-\gamma h/2} p_a^j + \xi_a^{j+1/2} \\ p_a^{j+1/2} &= p_a^* - \frac{h}{2} \frac{\partial V}{\partial q_a}(q_a^j) \\ q_a^{j+1} &= q_a^j + \frac{h}{m_a} p_a^{j+1/2} \\ p_a^{**} &= p_a^{j+1/2} - \frac{h}{2} \frac{\partial V}{\partial q_a}(q_a^{j+1}) \\ p_a^{j+1} &= e^{-\gamma h/2} p_a^{**} + \xi_a^{j+1}. \end{aligned}$$

Observe that this integrator is essentially the VV integrator sandwiched by two stochastic half-steps. During these half-steps, an impulse due to the random force is applied to the friction-dampened velocity.

A property that translates from the continuous setting to the discrete environment is the near preservation of the Boltzmann distribution. From the Fokker-Planck equation, the equilibrium phase-space distribution for the Langevin equations is Boltzmann so given an initial set of phase-space points satisfying the Boltzmann distribution, all subsequent snapshots of phase-space will also be Boltzmann. Since the stochastic updates are simply resampling the momenta according to the Boltzmann distribution and the inner VV integrator nearly preserves the Boltzmann distribution because the Hamiltionian is conserved to second-order, this stochastic integrator nearly preserves the Boltzmann distribution.

4.3 Extension to AVI

The formalism presented in Sec. 4.2 can be extended to handle multiple potentials, each with its own time step. Depending on how the stochastic action is approximated, a host of integrators can be derived from the variational framework. Here we will present three such approximations, beginning with one that borrows from the formulation seen in Sec. 2.5. To recall consider a system whose potential $V(q)$ can be written as the sum of K potentials:

$$V(q) = \sum_{k=1}^{K} V_k(q).$$

Then assign to each potential V_k a sequence of times $\{0 = t_k^0 < \ldots < t_k^{M_k} = T\}$. Additionally, we construct the sequence of all times in the system $\{\theta^0 < \theta^1 < \ldots < \theta^M\}$ by lumping together all potential times in a strictly increasing sequence. Denoting the position of the system at time θ^i by q^i, a discrete trajectory is the sequence of positions $\{q^0, \ldots, q^M\}$. For each time θ^i we define the set $\mathcal{K}(i)$ as:

$$\mathcal{K}(i) = \{k \mid \exists j,\ t_k^j = \theta^i\}.$$

For each $k \in \mathcal{K}(i)$, we can define:

$$h_k^{i+1/2} \overset{\text{def}}{=} t_k^{j+1} - t_k^j \text{ and } h_k^{i-1/2} \overset{\text{def}}{=} t_k^j - t_k^{j-1},$$

where $t_k^j = \theta^i$.

Combining elements from the discrete Lagrangians found in Eqs. (2.6) and (4.1) leads to the following:

$$\begin{aligned} L_{\mathrm{d}}^{\mathrm{s}}(q, \tilde{q}, i) = \\ e^{\gamma\theta^{i+1/2}} \left(\sum_{a=1}^{N} \frac{1}{2} m_a \Delta\theta^i \left\| \frac{\tilde{q}_a - q_a}{\Delta\theta^i} \right\|^2 - \sum_{k \in \mathcal{K}(i)} \frac{h_k^{i+1/2}}{2} V_k(q) - \sum_{k \in \mathcal{K}(i+1)} \frac{h_k^{i+1/2}}{2} V_k(\tilde{q}) \right) \\ + \sum_{a=1}^{N} \sum_{\alpha=1}^{d} \left(q_{a\alpha} e^{\gamma\theta^{i+1/2}} \xi_{a\alpha}^{i+1/2} + \tilde{q}_{a\alpha} e^{\gamma\theta^{i+1}} \xi_{a\alpha}^{i+1} \right) \quad (4.4) \end{aligned}$$

with $\Delta\theta^i = \theta^{i+1} - \theta^i$, $\theta^{i+1/2} = (\theta^i + \theta^{i+1})/2$, and $\xi_a^{i+1/2}$ and ξ_a^{i+1} are normal random vectors with mean $\vec{0}$ and covariance matrix given by $\left(m_a k_B T \left(1 - e^{-\gamma\Delta\theta^i}\right)\right) \delta_{lm}$. Using this stochastic discrete Lagrangian, the action sum is simply

$$S_{\mathrm{d}}^{\mathrm{s}} = \sum_{i=0}^{M-1} L_{\mathrm{d}}^{\mathrm{s}}(q^i, q^{i+1}, i).$$

The stochastic discrete Lagrangian $L_{\mathrm{d}}^{\mathrm{s}}$ is not a consistent approximation of the stochastic action during the time interval (θ^i, θ^{i+1}) but it is a consistent approximation of the stochastic action. The discrete Euler-Lagrange equations take the form:

$$\begin{aligned} m_a \dot{q}_a^{i+1/2} e^{\gamma\theta^{i+1/2}} - m_a \dot{q}_a^{i-1/2} e^{\gamma\theta^{i-1/2}} = \\ - \sum_{k \in \mathcal{K}(i)} \frac{h_k^{i-1/2} e^{\gamma\theta^{i-1/2}} + h_k^{i+1/2} e^{\gamma\theta^{i+1/2}}}{2} \frac{\partial V_k}{\partial q_a}(q^i) + e^{\gamma\theta^i} \xi_a^i + e^{\gamma\theta^{i+1/2}} \xi_a^{i+1/2} \quad (4.5) \end{aligned}$$

where

$$\dot{q}_a^{i+1/2} \overset{\text{def}}{=} \frac{q_a^{i+1} - q_a^i}{\theta^{i+1} - \theta^i}.$$

Eq. (4.3) defines the momenta $\{p^0, \ldots, p^M\}$, as

$$\begin{aligned}
p_a^i e^{\gamma\theta^i} &= m_a \dot{q}_a^{i-1/2} e^{\gamma\theta^{i-1/2}} - e^{\gamma\theta^{i-1/2}} \sum_{k\in\mathcal{K}(i)} \frac{h_k^{i-1/2}}{2} \frac{\partial V_k}{\partial q_a}(q^i) + e^{\gamma\theta^i} \xi_a^i \\
&= m_a \dot{q}_a^{i+1/2} e^{\gamma\theta^{i+1/2}} + e^{\gamma\theta^{i+1/2}} \sum_{k\in\mathcal{K}(i)} \frac{h_k^{i+1/2}}{2} \frac{\partial V_k}{\partial q_a}(q^i) - e^{\gamma\theta^{i+1/2}} \xi_a^{i+1/2}.
\end{aligned}$$

Letting $p_a^{i+1/2} = m_a \dot{q}_a^{i+1/2}$ the second momentum equation can be rearranged to give

$$p_a^{i+1/2} = e^{-\gamma\Delta\theta^i/2} p_a^i - \sum_{k\in\mathcal{K}(i)} \frac{h_k^{i+1/2}}{2} \frac{\partial V_k}{\partial q_a}(q^i) + \xi_a^{i+1/2}$$

while the first equation, replacing i with $i+1$, results in

$$p_a^{i+1} = e^{-\gamma\Delta\theta^i/2} \left(p_a^{i+1/2} - \sum_{k\in\mathcal{K}(i+1)} \frac{h_k^{i+1/2}}{2} \frac{\partial V_k}{\partial q_a}(q^{i+1}) \right) + \xi_a^{i+1}.$$

Hence the numerical integrator arising from the stochastic discrete Lagrangian (4.4) is

$$\begin{aligned}
p_a^* &= e^{-\gamma\Delta\theta^i/2} p_a^i + \xi_a^{i+1/2} \\
p_a^{i+1/2} &= p_a^* - \sum_{k\in\mathcal{K}(i)} \frac{h_k^{i+1/2}}{2} \frac{\partial V_k}{\partial q_a}(q^i) \\
q_a^{i+1} &= q_a^i + \frac{\Delta\theta^i}{m_a} p_a^{i+1/2} \\
p_a^{**} &= p_a^{i+1/2} - \sum_{k\in\mathcal{K}(i+1)} \frac{h_k^{i+1/2}}{2} \frac{\partial V_k}{\partial q_a}(q^{i+1}) \\
p_a^{i+1} &= e^{-\gamma\Delta\theta^i/2} p_a^{**} + \xi_a^{i+1}.
\end{aligned}$$

Observe that when all of time steps h_k are identical this integrator reduces to the integrator proposed in Sec. 4.2.

Another choice for the stochastic discrete Lagrangian is

$$
\begin{aligned}
L_{\mathrm{d}}^{\mathrm{s}}(q,\tilde{q},i) = & \\
& e^{\gamma\theta^{i+1/2}}\left(\sum_{a=1}^{N}\frac{1}{2}m_a\Delta\theta^i\left\|\frac{\tilde{q}_a-q_a}{\Delta\theta^i}\right\|^2\right) - e^{\gamma\theta^i}\sum_{k\in\mathcal{K}(i)}\frac{h_k^{i+1/2}}{2}V_k(q) \\
-e^{\gamma\theta^{i+1}}\sum_{k\in\mathcal{K}(i+1)}\frac{h_k^{i+1/2}}{2}V_k(\tilde{q}) + \sum_{a=1}^{N}\sum_{\alpha=1}^{d}&\left(q_{a\alpha}e^{\gamma\theta^{i+1/2}}\xi_{a\alpha}^{i+1/2}+\tilde{q}_{a\alpha}e^{\gamma\theta^{i+1}}\xi_{a\alpha}^{i+1}\right). \quad (4.6)
\end{aligned}
$$

The main difference between (4.4) and (4.6) is the time at which the exponential integrating factor is evaluated for the potential contributions V_k. By applying the trapezoidal rule to the product between the integrating factor and the potential V_k, the time is chosen to match the evaluation of the potential. This is contrasted by (4.4) where the midpoint of the time interval (θ^i, θ^{i+1}) was used for the integrating factor and the trapezoidal rule was only applied to the potential. Using the same variational machinery as for (4.4), the resulting integrator is

$$
\begin{aligned}
p_a^* &= p_a^i - \sum_{k\in\mathcal{K}(i)}\frac{h_k^{i+1/2}}{2}\frac{\partial V_k}{\partial q_a}(q^i) \\
p_a^{i+1/2} &= e^{-\gamma\Delta\theta^i/2}p_a^* + \xi_a^{i+1/2} \\
q_a^{i+1} &= q_a^i + \frac{\Delta\theta^i}{m_a}p_a^{i+1/2} \\
p_a^{**} &= e^{-\gamma\Delta\theta^i/2}p_a^{i+1/2} + \xi_a^{i+1} \\
p_a^{i+1} &= p_a^{**} - \sum_{k\in\mathcal{K}(i+1)}\frac{h_k^{i+1/2}}{2}\frac{\partial V_k}{\partial q_a}(q^{i+1}).
\end{aligned}
$$

Since the stochastic half-steps are now surrounded by the half-impulses from the potentials V_k, instead of vice versa, this integrator does not reduce to the integrator in Sec. 4.2 when all of the potentials are integrated with a single time step.

One situation where the stochastic integrators proposed so far might not be ideal is when the friction coefficient γ is large, i.e. when the dynamics are in the Brownian regime. As is evident from examining the integrators, the time steps for the stochastic updates are determined by the time steps chosen for the potentials V_k. If the potentials are smooth and/or slowly varying, the potential time steps h_k may be quite large and consequently $\Delta\theta^i$ will also be large for some values of i. Since the characteristic time scale of the thermal motion in Langevin dynamics is $1/\gamma$, the friction and random noise contributions will not be resolved accurately when large time steps are taken. One approach to circumvent this difficulty is to assign an independent time step h_0 for the integration of the stochastic contribution. Such an integrator can be obtained with the following approximation of the stochastic action:

$$\begin{aligned} S_{\mathrm{d}}^{\mathrm{s}}(\{q_i\}) = \sum_{i=0}^{M-1} e^{\gamma t_0^{j(i)}} & \left(\sum_{a=1}^{N} \frac{1}{2} m_a \Delta\theta^i \left\| \frac{q_a^{i+1} - q_a^i}{\Delta\theta^i} \right\|^2 \right) \\ & - \sum_{k=1}^{K} \sum_{l=0}^{M_k-1} (t_k^{l+1} - t_k^l) \frac{e^{\gamma t_0^{j(k,l)}} V_k(q^{k,l}) + e^{\gamma t_0^{j(k,l+1)}} V_k(q^{k,l+1})}{2} \\ & + \sum_{j=0}^{M_0-1} \sum_{a=1}^{N} \sum_{\alpha=1}^{d} \left(q_{a\alpha}^{0,j} e^{\gamma t_0^{j+1/2}} \xi_{a\alpha}^{j+1/2} + q_{a\alpha}^{0,j+1} e^{\gamma t_0^{j+1}} \xi_{a\alpha}^{j+1} \right) \end{aligned} \tag{4.7}$$

where $q^{k,l}$ is the position at time t_k^l and the times $t_0^{j(\cdot)}$ are defined as follows:

$$\begin{aligned} t_0^{j(i)} &= t_0^{j+1/2} \mid \theta^i \in [t_0^j, t_0^{j+1}) \\ t_0^{j(k,l)} &= t_0^{j+1/2} \mid t_k^l \in [t_0^j, t_0^{j+1}) \\ t_0^{j(k,l+1)} &= t_0^{j+1/2} \mid t_k^{l+1} \in (t_0^j, t_0^{j+1}]. \end{aligned}$$

Letting the scaled momemtum at time θ^i be given by $p_a^i e^{\gamma t_0^{j(i)}}$ and applying the discrete variational principle to (4.7), the integrator for the time interval (θ^i, θ^{i+1}) where

$\theta^i \neq t_0^j$ and $\theta^{i+1} \neq t_0^j$ for all j is exactly the AVI integrator:

$$\begin{aligned}
p_a^{i+1/2} &= p_a^i - \sum_{k \in \mathcal{K}(i)} \frac{h_k^{i+1/2}}{2} \frac{\partial V_k}{\partial q_a}(q^i) \\
q_a^{i+1} &= q_a^i + \frac{\Delta\theta^i}{m_a} p_a^{i+1/2} \\
p_a^{i+1} &= p_a^{i+1/2} - \sum_{k \in \mathcal{K}(i+1)} \frac{h_k^{i+1/2}}{2} \frac{\partial V_k}{\partial q_a}(q^{i+1}).
\end{aligned}$$

When $\theta^i = t_0^j$ for some j then the first half-step impulse is preceded by

$$p_a^* = e^{-\gamma h_0/2} p_a^i + \xi_a^{j+1/2}$$

and when $\theta^{i+1} = t_0^j$ for some j then the second half-step impulse is followed by

$$p_a^{**} = e^{-\gamma h_0/2} p_a^{i+1} + \xi_a^{j+1}.$$

Note that when all of time steps, including h_0, are replaced with a single time step, this integrator reduces to the integrator in Sec. 4.2.

4.4 Application of SVI to a Peptide

To investigate the consistency of the last of the three integrators proposed in the previous section we will apply it to a molecular dynamics simulation of a solvated 17-residue α-helix peptide under periodic boundary conditions. Particle mesh Ewald (PME) was used to integrate the electrostatic potential, with the Ewald coefficient equal to 0.646. Two time steps were adopted: a variable time step h is assigned to the reciprocal-space portion of the Ewald sum while the time step for all other potentials, including the stochastic contribution from Langevin dynamics, is 1 fs. A friction coefficient of 5 ps^{-1} was used and the temperature of the system was set at 300 K. Beginning with a 20-member ensemble of the solvated peptide drawn from the Boltzmann distribution, each ensemble member was simulated for 100 fs and its final configurated was recorded. Using $h = 1$ as the reference solution, the averaged trajectory error for time step h is defined as

$$\varepsilon_h = \left[\frac{1}{3N_{\text{atoms}}N_{\text{ens}}} \sum_{j=1}^{N_{\text{ens}}} \sum_{a=1}^{N_{\text{atoms}}} ||q_{a,h}^{(j)} - q_{a,\text{ref}}^{(j)}||_2^2 \right]^{1/2}$$

where N_{atoms} is the number of atoms, N_{ens} is the number of ensemble members, $q_{a,h}^{(j)}$ is the position vector of atom a for the j-th ensemble member at 100 fs obtained by using time step h for the reciprocal-space part of the Ewald sum, and $q_{a,\text{ref}}^{(j)}$ is the respective position vector when $h = 1$. Taking values of h from 1.2 to 3 fs in increments of 0.2 fs, the error ε_h was computed and the results are shown in Fig. 4.1. From the plot it can be concluded that this particular stochastic integrator exhibits roughly first-order convergence with respect to h.

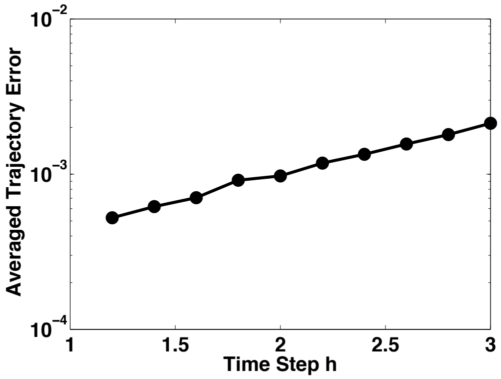


Figure 4.1: The averaged trajectory error as a function of the time step used for the reciprocal-space contribution of the Ewald sum. The time step for all other potentials, including the stochastic contribution from Langevin dynamics, was taken to be 1 fs. The integrator exhibits roughly first-order convergence.

4.5 Summary

We showed that stochastic variational integrators can be derived from the variational framework by modifying the action integral. However depending on how this stochastic action is discretized, a variety of integrators can be obtained. In this dissertation, we only present numerical results for one of the integrators. Additional analysis and investigation is required to determine the benefits and disadvantages of each integrator and to see if there exists one integrator that is superior to the others. Furthermore it remains to be seen how accurately the proposed integrators preserve the Boltzmann distribution, an important property of Langevin dynamics.

Chapter 5

Background on the Fast Multipole Method

The fast multipole method (FMM) is a technique for calculating sums of the form

$$f(x_i) = \sum_{j=1}^{N} K(x_i, y_j)\sigma_j, \quad i = 1, \dots, N \tag{5.1}$$

in $O(N)$ operations with a controllable error ε. Historically, Greengard and Rokhlin [36] first developed a technique for the kernel $K(x, y) = 1/r$ ($r = |x - y|$) based on Legendre polynomials and spherical harmonics. The technique was later extended to the oscillatory kernel e^{ikr}/r. Both these approaches require approximations of $K(x, y)$ for $|x - y|$ sufficiently large (in a sense which can be made precise) which are typically obtained using known analytical expansions. In this chapter we will focus on the $1/r$ kernel and begin by discussing how a low-rank approximation of $1/r$ can be constructed for large r. Next we proceed to show how a $O(N \log N)$ fast summation method can be derived from this low-rank approximation. Finally a detailed description of the $O(N)$ FMM will be provided.

5.1 Constructing A Low-Rank Approximation

Consider sums of the form (5.1) where x_i are observation points, σ_j are the sources, y_j are the locations of the sources, and N is the number of observation points and sources. These sums appear in many applications such as N-body problems and integral equations in electromagnetics and acoustics. A direct calculation of this sum has a $O(N^2)$ complexity resulting from the multiplication of a $N \times N$ matrix $K_{ij} = K(x_i, y_j)$ with the N-vector of sources $\{\sigma_j\}$. Since the number of observation points and sources is often very large computing the sum directly is intractable. An approach for improving the efficiency of this computation involves the use of a low-rank approximation of the kernel:

$$K(x, y) \approx \sum_{l=1}^{n} u_l(x) v_l(y). \tag{5.2}$$

A fast summation method can be created by substituting the low-rank approximation (5.2) into the sum (5.1):

$$f(x_i) \approx \sum_{l=1}^{n} u_l(x_i) \sum_{j=1}^{N} v_l(y_j)\sigma_j. \tag{5.3}$$

The outline for the method given by Eq. (5.3) is as follows:

1. First transform the sources using the basis functions v_l:

$$W_l = \sum_{j=1}^{N} v_l(y_j)\sigma_j, \quad l = 1, \ldots, n.$$

2. Then compute $f(x)$ at each observation point x_i using the basis functions u_l:

$$f(x_i) \approx \sum_{l=1}^{n} u_l(x_i) W_l, \quad i = 1, \ldots, N.$$

The computational cost of each step is $O(nN)$ hence the fast summation method proposed above scales as $O(2nN)$. When $n \ll N$ this is a significant reduction from the $O(N^2)$ scaling of the direct calculation.

A discussion on the derivation of a series expansion of the kernel $K(x, y) = 1/r$ in three dimensions is given in [add citation], whose details will be summarized here. We are interested in computing the potential at the observation point x due to a source at y. Let the spherical coordinates of these points be given by $x = (r, \theta, \phi)$ and $y = (\rho, \alpha, \beta)$. If we denote the angle between the vectors x and y by γ and let $\tilde{r}$ be the distance between x and y then the kernel $1/\tilde{r}$ can be expressed as

$$\frac{1}{\tilde{r}} = \sum_{l=0}^{\infty} \frac{\rho^l}{r^{l+1}} P_l(\cos\gamma) \tag{5.4}$$

where P_l is the Legendre polynomial of degree l. This series only converges if $r > \rho$ so this can be interpreted as a far-field expansion of the $1/\tilde{r}$ kernel. Since the angle γ is a function of both x and y this is not a low-rank approximation of the kernel in the form of (5.2). However x and y can be separated by appealing to the addition theorem for Legendre polynomials which expands the Legendre polynomial as a sum of spherical harmonics

$$P_l(\cos\gamma) = \sum_{m=-l}^{l} Y_l^m(\theta, \phi) Y_l^{-m}(\alpha, \beta) \tag{5.5}$$

where

$$Y_l^m(\theta, \phi) = \sqrt{\frac{(l - |m|)!}{(l + |m|)!}}\; P_l^{|m|}(\cos\theta) e^{im\phi}$$

and P_l^m are the associated Legendre functions which may be defined by the Rodrigues' formula

$$P_l^m(z) = (-1)^m (1 - z^2)^{m/2} \frac{d^m}{dx^m} P_l(z).$$

Substituting Eq. (5.5) into (5.4) gives a low-rank approximation of the kernel $1/\tilde{r}$:

$$\frac{1}{\tilde{r}} \approx \sum_{l=0}^{n} \sum_{m=-l}^{l} \frac{1}{r^{l+1}} Y_l^m(\theta, \phi)\; \rho^l Y_l^{-m}(\alpha, \beta). \tag{5.6}$$

Now consider a set of N sources located at $y_j = (\rho_j, \alpha_j, \beta_j)$ with $\rho_j < a$ for $j = 1, \ldots, N$ and N observation points $x_i = (r_i, \theta_i, \phi_i)$ with $r_i > a$ for $i = 1, \ldots, n$. Then the power series in Eq. (5.4) converges for all pairwise interactions and hence Eq. (5.6) can be used for approximating the sum (5.1):

$$f(x_i) \approx \sum_{l=0}^{n} \sum_{m=-l}^{l} \frac{1}{r_i^{l+1}} Y_l^m(\theta_i, \phi_i) M_l^m \tag{5.7}$$

with

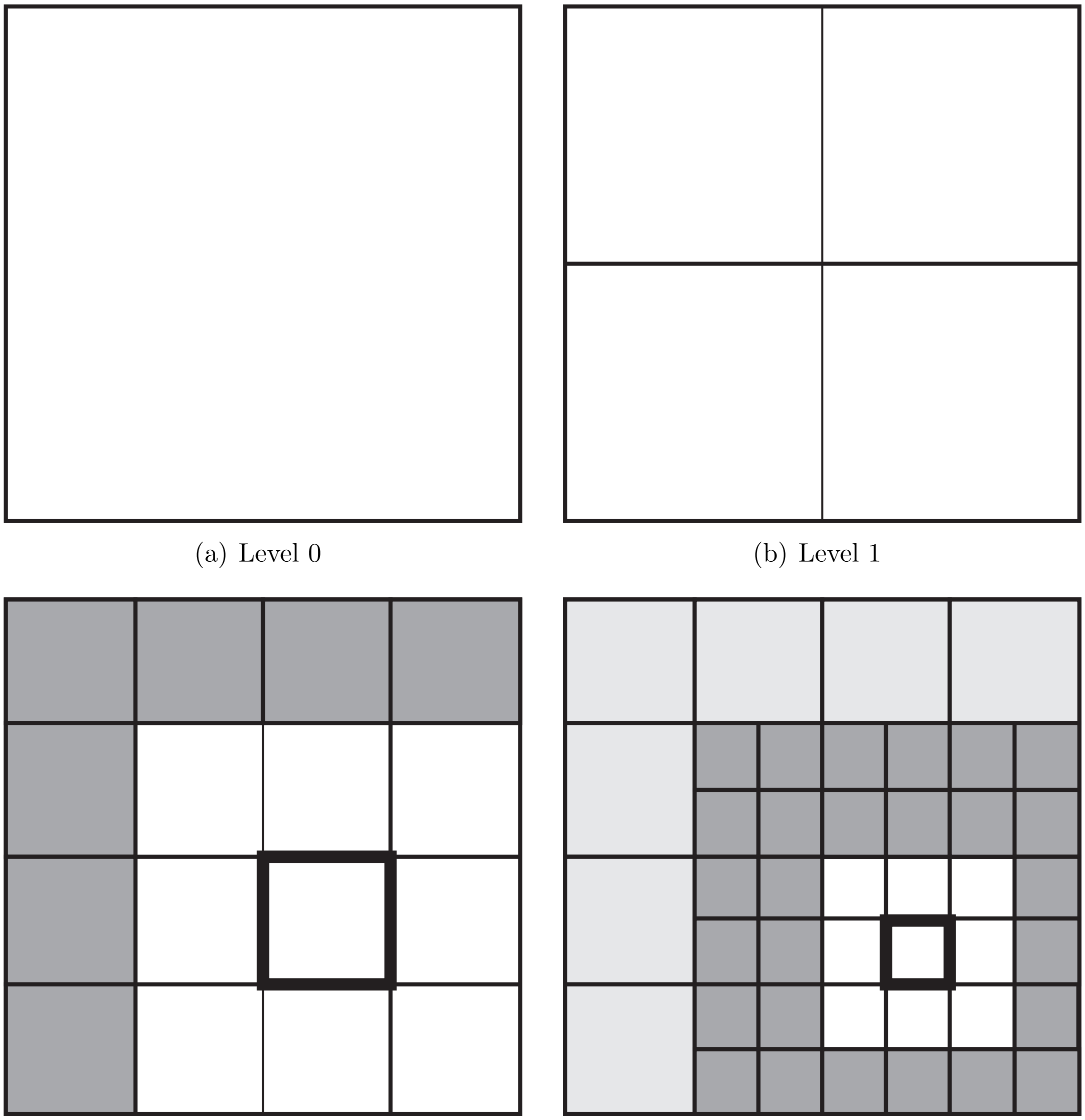
$$M_l^m = \sum_{j=1}^{N} \rho_j^l Y_l^{-m}(\alpha_j, \beta_j) \sigma_j. \tag{5.8}$$

Eq. (5.7) is commonly referred to as the multipole expansion and (5.8) are the corresponding multipole coefficients.

5.2 An $O(N \log N)$ Fast Summation Method

Previously we discussed how to construct a low-rank approximation of the kernel $1/r$ that is valid when the observation points are separated from the sources. However in many situations this separation does not exist. For example consider a set of point charges in a cubic domain and we are interested in evaluating the potential at each point charge due to the other charges. Since the set of observation points and the set of sources are identical and hence not separated, Eq. (5.7) cannot be applied directly to this system. One solution is to subdivide the computational cell recursivey to produce distinct groups of observation points and sources that are separated such that the far-field expansion is valid. For simplicity we will go to two dimensions and take a square as our computational domain to illustrate how this procedure works. The strategy discussed hereafter can be easily extended to three dimensions.

Let refinement level 0 correspond to the entire computational domain (Fig. 5.1(a)). At this level, the far-field approximation cannot be applied. Therefore we subdivide the square cell into four smaller subcells whose edge length is half that of the original square and refer to this as refinement level 1 (Fig. 5.1(b)). Since the four subcells are



(a) Level 0

(b) Level 1

(c) Level 2

(d) Level 3

Figure 5.1: Levels in the FMM tree for 2-D domain. On levels 0 and 1, none of the cells are well-separated so the far-field approximation cannot be applied. However by recursively subdividing, clusters of observation points and source can be separated. For example, on level 2 (c) interactions between observation points in the highlighted cell and sources in the shaded cells can be computed using the multipole expansion. In order to handle sources in the adjacent unshaded cells, a further subdivision is done (d). The light shaded cells have been accounted for on the previous level so only the dark shaded cells need to be treated with the multipole expansion. If this level corresponds to the finest subdivision, then interactions due to the sources in the unshaded cells are computed directly.

adjacent to one another, the observation points in any subcell is not separated from the sources in any other subcell. As a result we will repeat this subdivision again to form refinement level 2 with 16 subcells. Before proceeding we will introduce some terminology that will be helpful in the description to follow. Two cells are **near neighbors** if they are on the same refinement level and share at least one vertex. From this definition, a cell is a near neighbor of itself. Next two cells are **well-separated** if they are on the same refinement level and not near neighbors. For square or cubic cells, a sufficient condition for well-separateness is that two cells are separated by at least one cell in each dimension. Finally the **interaction list** of cell i is a list consisting of all children of the near neighbors of cell i's parent that are well-separated from cell i. Referring to Fig. 5.1(c) the interaction list of the highlighted cell contains the seven shaded cells. Since these cell-cell interactions are well-separated we can now use the multipole expansion to compute the potential at observation points located in the highlighted cell due to the sources in each of the cells in the interaction list. The remaining unshaded cells are near neighbors of the highlighted cell. In order to apply the far-field expansion for sources located in the near neighbor cells, the refinement is continued recursively and the contributions from these sources are handled at subsequent refinement levels (e.g. Fig. 5.1(d)). If the density of the observation points and sources is relatively uniform, then the number of levels is taken to be roughly $\log N$ to ensure that most of the subcells with the finest refinement are non-empty. For subcells on the last refinement level, the remaining interactions due to sources in its near neighbors are computed directly. This recursive subdivision gives rise to the FMM tree where each node corresponds to a subcell and an edge represents a parent-child relationship. Letting the root node be the entire computational domain, each subsequent level of the tree represents the next refinement level. In two dimensions the FMM tree is a quadtree while in three dimensions we have an octree. From this tree structure the near neighbors and interaction list of any cell can be easily determined.

Since the computation of the multipole expansion (5.7) on each level of the FMM tree is $O(N)$ because there are N observation points and N sources, the total cost of the fast summation method is $O(N \log N)$. In order to obtains an $O(N)$ complexity, operations on the observation points and sources can only be done on one level instead of on all refinement levels. The fast multipole method achieves this by working with the observation points and sources only on the finest refinement level and passes this information to the other levels in the FMM tree by employing analytical results for expansions in spherical harmonics.

5.3 The Fast Multipole Method

The fast multipole method is comprised of two passes through the FMM tree: an upward pass and a downward pass. To begin, the multipole coefficients (5.8) are computed for all cells corresponding to the finest refinement level. In order to calculate the multipole coefficients for the coarser levels, we will use an analytical formula that describes how a multipole expansion can be translated.

Suppose that for a set of sources located in a sphere of radius a with center $y = (\rho, \alpha, \beta)$ and that for observation points $x = (r, \theta, \phi)$ outside of this sphere, the potential is given by the multipole expansion

$$f(x) = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} \frac{O_l^m}{(r')^{l+1}} Y_l^m(\theta', \phi') \tag{5.9}$$

where $x - y = (r', \theta', \phi')$. Then for any x outside of the sphere of radius $a + \rho$ centered at y

$$f(x) = \sum_{j=0}^{\infty} \sum_{k=-j}^{j} \frac{M_j^k}{r^{l+1}} Y_j^k(\theta, \phi) \tag{5.10}$$

with

$$M_j^k = \sum_{l=0}^{j} \sum_{m=-l}^{l} \frac{O_{j-l}^{k-m} \; i^{|k|-|m|-|k-m|} \; A_l^m \; A_{j-l}^{k-m} \; \rho^l \; Y_l^{-m}(\alpha, \beta)}{A_j^k} \tag{5.11}$$

and

$$A_l^m = \frac{(-1)^l}{\sqrt{(n-m)!(n+m)!}} \tag{5.12}$$

By using Eqs. (5.10) and (5.11) the multipole expansion for a parent cell can be computed from those of its children cells. This operation is known as the multipole-to-multipole (M2M) operation.

After forming all of the multipole expansions on the coarser levels recursively, we now shift our attention to the downward pass. To begin we compute the local expansion, a series expansion that is only valid within a certain radius, with the help of another analtyical formula.

Suppose that for a set of sources located inside the sphere of radius a centered at $y = (\rho, \alpha, \beta)$ with $\rho > (c+1)a$ and $c > 1$. Then the multipole expansion (5.9) for this set of sources converges inside the sphere with radius a centered at the origin. The potential due to the sources evaluated at a point $x = (r, \theta, \phi)$ inside this sphere is given by the local expansion

$$f(x) = \sum_{j=0}^{\infty} \sum_{k=-j}^{j} L_j^k \; Y_j^k(\theta, \phi) \; r^j \tag{5.13}$$

with

$$L_j^k = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} \frac{O_l^m \; i^{|k-m|-|k|-|m|} \; A_l^m \; A_j^k \; Y_{j+l}^{m-k}(\alpha, \beta)}{(-1)^l \; A_{j+l}^{m-k} \; \rho^{j+l+1}} \tag{5.14}$$

and A_l^m is given by (5.12). By using Eqs. (5.13) and (5.14) the local expansion for each subcell in the FMM tree can be constructed. This operation is known as the multipole-to-local (M2L) operation.

Since the local expansion for a cell describes only the potential due to sources located in cells listed the interaction list, information regarding the potential due to more distantly-located sources needs to be obtained from the parent cell. This can be done by first translating the local expansion of the parent and then adding the result to the local expansion of the children cells. Once again there exists an analytical formula that performs this action.

Letting $y = (\rho, \alpha, \beta)$ be the origin of a local expansion

$$f(x) = \sum_{l=0}^{n} \sum_{m=-n}^{n} O_l^m \; Y_l^m(\theta', \phi') \; (r')^l$$

where $x = (r, \theta, \phi)$ and $x - y = (r', \theta', \phi')$ then

$$f(x) = \sum_{j=0}^{n} \sum_{k=-j}^{j} L_j^k \; Y_j^k(\theta, \phi) \; r^j \tag{5.15}$$

where

$$L_j^k = \sum_{l=j}^{n} \sum_{m=-l}^{l} \frac{O_l^m \; i^{|m|-|m-k|-|k|} \; A_{l-j}^{m-k} \; A_j^k \; Y_{l-j}^{m-k}(\alpha, \beta) \; \rho^{l-j}}{(-1)^{l+j} \; A_l^m} \tag{5.16}$$

with A_l^m given by (5.12). By using Eqs. (5.15) and (5.16) the local expansion for a parent cell can be passed and added to its children cells. This operation is known as the local-to-local (M2M) operation. After passing down and adding local expansions recursively, the final step to the downward pass is to add the contributions to the potential from sources located in the near neighbors of the subcells on the finest refinement level. From these two passes observe that the observation points and sources are only used once when working on the finest level of the FMM tree. As a result, this approach, called the fast multipole method, gives the desired $O(N)$ complexity.

Chapter 6

The Black-Box Fast Multipole Method

A new $O(N)$ fast multipole formulation has been proposed for non-oscillatory kernels. This algorithm is applicable to kernels $K(x, y)$ which are only known numerically, that is their numerical value can be obtained for any (x, y). This is quite different from many fast multipole methods which depend on analytical expansions of the far field behavior of K, for $|x - y|$ large. Other "black-box" or "kernel-independent" fast multipole methods have been devised. Our approach has the advantage of requiring a small pre-computation time even for very large systems, and uses the minimal number of coefficients to represent the far field, for a given L^2 tolerance error in the approximation. This technique can be very useful for problems where the kernel is known analytically but is quite complicated, or for kernels which are defined purely numerically.

6.1 Introduction

Extensions to general kernels are possible, for example using Taylor expansions. Fewer methods exist which allow building a fast $O(N)$ method using only numerical values of K, that is without requiring approximations based on analytical expansions. These techniques are often based on wavelet decompositions [18, 2], singular value decompositions [33], or other schemes [13].

In Gimbutas et al. [33], a scheme based on singular value decompositions (SVD) is used. Using the usual tree decomposition of the domain, they denote by Y^b a cluster at level l and by Z^b the union of Y^b and its nearest neighbor clusters at level l. Then X^b is defined as the complement of Z^b. The kernel $K(x, y)$ can then be decomposed using a continuous SVD:

$$K(x,y) \approx \sum_{l=1}^{n} s_l \, u_l(x) \, v_l(y)$$

where $y \in Y^b$ and $x \in X^b$. This low-rank approximation can be extended to produce multipole-to-multipole, local-to-local and multipole-to-local operators. The advantage of this technique is that the SVD guarantees an optimal compression. Therefore the number of multipole coefficients that one operates with is minimal for a given approximation error in the L^2 norm. The drawback of this approach is the cost of pre-computing the SVD of K which can be very expensive for large 3-D domains.

Interpolation techniques can be used to construct fast multipole methods. This approach has not attracted a lot of attention but a few papers have used interpolation techniques (e.g., Chebyshev polynomials) in various ways as part of constructing fast methods [30, 27, 28]. The reference [26] discusses an idea similar to that discussed in this chapter, with some differences, including the fact that the multipole and local expansions are treated differently, whereas our scheme is more "symmetrical" and treats them in a similar way. In addition, [26] focuses on a one-dimensional FMM with the kernel $1/x$, which is required by their fast algorithm for interpolation, differentiation and integration.

The basic idea of an interpolation-based FMM is as follows. If we let $w_l(x)$ denote the interpolating functions, then:

$$K(x,y) \approx \sum_{lm} K(x_l, y_m) w_l(x) w_m(y)$$

which is a low-rank approximation. This works for any interpolation scheme. The advantage of this type of approach is that it requires minimal pre-computing. In addition, the only input required is the ability to evaluate K at various points. No kernel-dependent analytical expansion is required. The drawback is that the number of expansion terms (indices l and m in the sum above) is in general sub-optimal for a given tolerance ε.

In this chapter we propose a new approach which essentially combines these two ideas. A Chebyshev interpolation scheme is used to approximate the far-field behavior of $K(x, y)$, i.e., when $|x - y|$ large. This leads to an efficient low-rank representation for non-oscillatory kernels. The multipole-to-local (M2L) operator then consists in evaluating the field due to particles located at Chebyshev nodes. This operation can be done efficiently using an SVD. This makes the scheme optimal since the M2L step is by far the most expensive. A key point is that the SVD needs to be computed only "locally". More specifically, given a tolerance ε, if the kernel is translation-invariant, i.e., of the form $K(x - y)$, then the cost of pre-computing the SVD is $O(\ln N)$; otherwise the pre-computing cost is $O(N)$. This is a true pre-computation since this calculation is independent of the location of the sources y_j and observation points x_i, and only depends on the desired accuracy.

We begin by discussing interpolation methods and in particular Chebyshev polynomials, which have several desirable properties. Then we explain how one can construct on $O(N)$ fast method from this interpolation scheme, and how it can be further accelerated using singular value decompositions. Finally some numerical results illustrate the accuracy and efficiency of the method.

6.2 Using Chebyshev Polynomials as an Interpolation Basis

A low-rank approximation of the kernel $K(x, y)$ can be constructed by introducing an interpolation scheme. To begin consider a function $g(x)$ on the closed interval $[-1, 1]$. An n-point interpolant that approximates $g(x)$ can be expressed as

$$p_{n-1}(x) = \sum_{l=1}^{n} g(x_l) w_l(x) \tag{6.1}$$

where $\{x_l\}$ are the n interpolation nodes and $w_l(x)$ is the interpolating function corresponding to the node x_l. For example if the functions $w_l(x)$ are taken to be the Lagrange polynomials then $p_{n-1}(x)$ is a $(n-1)$-degree polynomial approximation of $g(x)$. Eq. (6.1) can be used to approximate the kernel $K(x, y)$ by first fixing the variable y and treating $K(x, y)$ as a function of x:

$$K(x, y) \approx \sum_{l=1}^{n} K(x_l, y) w_l(x).$$

Now noting that $K(x_l, y)$ is a function of y the interpolation formula (6.1) can be applied again to give

$$K(x, y) \approx \sum_{l=1}^{n} \sum_{m=1}^{n} K(x_l, y_m) w_l(x) w_m(y) \tag{6.2}$$

which is a low-rank representation of the kernel $K(x, y)$ with

$$\begin{aligned} u_l(x) &= w_l(x) \\ v_l(y) &= \sum_{m=1}^{n} K(x_l, y_m) w_m(y). \end{aligned}$$

Although any interpolation scheme can be used to construct a low-rank approximation, the Chebyshev polynomials will serve as the interpolation basis along with their roots as the interpolation nodes. Before justifying this selection we begin by recalling some properties of Chebyshev polynomials.

The first-kind Chebyshev polynomial of degree n, denoted by $T_n(x)$, is defined by the relation

$$T_n(x) = \cos(n\theta), \quad \text{with } x = \cos\theta.$$

The domain of $T_n(x)$ is the closed interval $[-1, 1]$. $T_n(x)$ has n roots located at

$$\bar{x}_m = \cos\theta_m = \cos\left(\frac{(2m-1)\pi}{2n}\right), \quad m = 1, \ldots, n$$

and $n+1$ extrema located at

$$\bar{x}'_m = \cos\theta'_m = \cos\left(\frac{m\pi}{n}\right), \qquad \text{with } T_n(\bar{x}'_m) = (-1)^m, \quad m = 0, \ldots, n.$$

The set of roots $\{\bar{x}_m\}$ is commonly referred to as the Chebyshev nodes.

One advantage of using Chebyshev nodes is the stability of the interpolation scheme. While a scheme using equally-spaced nodes on the $[-1,1]$ interval to interpolate a function $g(x)$ suffers from Runge's phenomenon and does not converge uniformly as the number of nodes n becomes large, Chebyshev interpolation ensures uniform convergence with minimal restrictions on $g(x)$. Another benefit afforded by interpolating at the Chebyshev nodes is the near-minimax approximation of $g(x)$ which gives a uniform approximation error across the interval $[-1,1]$. This can be contrasted with the error behavior in the regular multipole expansion of the Laplacian kernel $1/r$ using spherical harmonics. Similar to a Taylor series expansion, the regular multipole expansion is very accurate around the center of the interval but suffers from larger errors near the endpoints. Therefore to ensure a specified accuracy across the entire interval a high-order interpolant is needed in order to adequately resolve the endpoints. The uniform error distribution of Chebyshev interpolation allows for the use of fewer interpolation nodes to achieve a given accuracy and is nearly optimal in the minimax sense.

Using the Chebyshev nodes of $T_n(x)$ as the interpolation nodes, the approximating polynomial $p_{n-1}(x)$ to the function $g(x)$ can be expressed as a sum of Chebyshev polynomials

$$p_{n-1}(x) = \sum_{k=0}^{n-1} c_k T_k(x)$$

where

$$c_k = \begin{cases} \frac{2}{n}\sum_{l=1}^{n} g(\bar{x}_l) T_k(\bar{x}_l) & \text{if } k > 0 \\ \frac{1}{n}\sum_{l=1}^{n} g(\bar{x}_l) & \text{if } k = 0 \end{cases}$$

and $\bar{x}_l$ are the roots of $T_n(x)$. By rearranging the terms in the sum, $p_{n-1}(x)$ can be written in the form of (6.1):

$$p_{n-1}(x) = \sum_{l=1}^{n} g(\bar{x}_l) S_n(\bar{x}_l, x)$$

where

$$S_n(x, y) = \frac{1}{n} + \frac{2}{n} \sum_{k=1}^{n-1} T_k(x) T_k(y).$$

The rate of convergence of $p_n(x)$ to $g(x)$ is given by two results from Chebyshev approximation theory [67]. First if $g(x)$ has $m + 1$ continuous derivatives on $[-1, 1]$, then the pointwise approximation error for all $x \in [-1, 1]$ is

$$|g(x) - p_n(x)| = O(n^{-m}).$$

Second if $g(x)$ can be extended to a function $g(z)$, where z is a complex variable, which is analytic within a simple closed contour C that encloses the point x and all the roots of the Chebyshev polynomial $T_{n+1}(x)$ then the interpolating polynomial $p_n(x)$ can be written as

$$p_n(x) = \frac{1}{2\pi i} \int_C \frac{[T_{n+1}(z) - T_{n+1}(x)] g(z)}{T_{n+1}(z)(z - x)} dz$$

and its error is

$$g(x) - p_n(x) = \frac{1}{2\pi i} \int_C \frac{T_{n+1}(x) g(z)}{T_{n+1}(z)(z - x)} dz.$$

Moreover if $g(x)$ extends to an analytic function within the ellipse E_r given by the locus of points $\frac{1}{2}(r \exp(i\theta) + r^{-1} \exp(-i\theta))$ (for some $r > 1$ and as θ varies from 0 to 2π) and $|g(z)| \leq M$ at every point z on E_r then for every real $x \in [-1, 1]$ the approximating polynomial $p_n(x)$ exhibits spectral convergence:

$$|g(x) - p_n(x)| \leq \frac{(r + r^{-1}) M}{(r^{n+1} + r^{-(n+1)})(r + r^{-1} - 2)}.$$

This exponential accuracy is yet another desirable aspect of using Chebyshev polynomials for the interpolation basis.

Identifying $S_n(\bar{x}_l, x)$ as the interpolating function for the node $\bar{x}_l$, it follows from Eq. (6.2) that a low-rank approximation of the kernel $K(x,y)$ using Chebyshev polynomials is given by

$$K(x,y) \approx \sum_{l=1}^{n} \sum_{m=1}^{n} K(\bar{x}_l, \bar{y}_m) S_n(\bar{x}_l, x) S_n(\bar{y}_m, y). \tag{6.3}$$

Substituting this expression into Eq. (5.1) and changing the order of summation we have

$$\begin{aligned} f(x_i) &= \sum_{j=1}^{N} K(x_i, y_j)\sigma_j \\ &\approx \sum_{j=1}^{N} \left[\sum_{l=1}^{n} \sum_{m=1}^{n} K(\bar{x}_l, \bar{y}_m) S_n(\bar{x}_l, x_i) S_n(\bar{y}_m, y_j) \right] \sigma_j \\ &= \sum_{l=1}^{n} S_n(\bar{x}_l, x_i) \sum_{m=1}^{n} K(\bar{x}_l, \bar{y}_m) \sum_{j=1}^{N} \sigma_j S_n(\bar{y}_m, y_j). \end{aligned}$$

From this decomposition a fast summation method using Chebyshev interpolation can be constructed.

1. First compute the weights at the Chebyshev nodes $\bar{y}_m$ by anterpolation:

$$W_m = \sum_{j=1}^{N} \sigma_j S_n(\bar{y}_m, y_j), \quad m = 1, \dots, n$$

2. Next compute $f(x)$ at the Chebyshev nodes $\bar{x}_l$:

$$f(\bar{x}_l) = \sum_{m=1}^{n} W_m K(\bar{x}_l, \bar{y}_m), \quad l = 1, \dots, n$$

3. Last compute $f(x)$ at the observation points x_i by interpolation:

$$f(x_i) = \sum_{l=1}^{n} f(\bar{x}_l) S_n(\bar{x}_l, x_i), \quad i = 1, \dots, N$$

The computational cost of steps 1 and 3 are both $O(nN)$ while step 2 is $O(n^2)$, hence for $n \ll N$ the algorithm scales like $O(2nN)$.

The decomposition above can be extended to include kernels $K(x, y)$ that are defined over arbitrary rectangular domains $[a, b] \times [c, d]$ by mapping back to the square $[-1, 1] \times [-1, 1]$ via linear transformation. In addition this fast summation method can be extended to higher dimensions by taking a tensor product of the interpolating functions S_n, one for each dimension. For example consider a 3-D kernel $K(\mathbf{x}, \mathbf{y})$ where $\mathbf{x} = (x_1, x_2, x_3)$ and $\mathbf{y} = (y_1, y_2, y_3)$. The low-rank approximation of the kernel $K(\mathbf{x}, \mathbf{y})$ using Chebyshev polynomials can be expressed as

$$K(\mathbf{x}, \mathbf{y}) \approx \sum_{\mathbf{l}} \sum_{\mathbf{m}} K(\bar{\mathbf{x}}_{\mathbf{l}}, \bar{\mathbf{y}}_{\mathbf{m}}) R_n(\bar{\mathbf{x}}_{\mathbf{l}}, \mathbf{x}) R_n(\bar{\mathbf{y}}_{\mathbf{m}}, \mathbf{y}) \tag{6.4}$$

where

$$R_n(\mathbf{x}, \mathbf{y}) = S_n(x_1, y_1) S_n(x_2, y_2) S_n(x_3, y_3)$$

and $\bar{\mathbf{x}}_{\mathbf{l}} = (\bar{x}_{l_1}, \bar{x}_{l_2}, \bar{x}_{l_3})$ and $\bar{\mathbf{y}}_{\mathbf{m}} = (\bar{y}_{m_1}, \bar{y}_{m_2}, \bar{y}_{m_3})$ are 3-vectors of Chebyshev nodes with $l_i, m_i \in \{1, \ldots, n\}$.

6.3 A Black-Box FMM with Chebyshev Interpolation

In the previous section a fast summation method was constructed for continuous kernels based on a low-rank approximation using Chebyshev polynomials. However if the kernel contains discontinuities in its domain, e.g., the Laplacian kernel $1/|x - y|$, this low-rank representation is not applicable. In order for the low-rank approximation (6.3) to accurately represent the kernel, the observation and source intervals need to be well-separated. Hence (6.3) can be thought of as a far-field approximation of the kernel $K(x, y)$. Local interactions involving observation points and sources in non-well-separated intervals can also be computed with the far-field approximation by subdividing the intervals. On this refined scale, interactions between well-separated observation points and sources can be treated by (6.3). Applying this refinement recursively produces a multilevel fast summation method. A black-box fast multipole method (bbFMM) can be constructed by combining this multilevel scheme with the FMM tree structure. Our method is a black box in the sense that the functional form of the low-rank approximation (6.3) is independent of the kernel. Let the root level of the tree (level 0) be the computational interval containing all observation points and sources. The algorithm for a κ-level 1-D bbFMM is as follows:

1. First compute the weights at the Chebyshev nodes $\bar{y}_m$ for all subintervals I on level κ by anterpolation:

$$W_m^{I,\kappa} = \sum_{\substack{y_j \text{ in} \\ \text{subinterval } I}} \sigma_j S_n(\bar{y}_m^{I,\kappa}, y_j), \quad m = 1, \dots, n$$

2. Next compute the weights at the Chebyshev nodes $\bar{y}_m$ for all subintervals I on level k by recursion, $\kappa - 1 \geq k \geq 0$ (M2M):

$$W_m^{I,k} = \sum_{\substack{\bar{y}_m^{J,k+1} \text{ in child} \\ \text{intervals of } I}} W_m^{J,k+1} S_n(\bar{y}_m^{I,k}, \bar{y}_m^{J,k+1}), \quad m = 1, \dots, n$$

3. Then calculate the far-field contribution at the Chebyshev nodes $\bar{x}_l$ for all subintervals J in the interaction list of I on level k, $0 \leq k \leq \kappa$ (M2L):

$$g(\bar{x}_l^{I,k}) = \sum_{\substack{\bar{y}_m^{J,k} \text{ in interaction} \\ \text{list of } I}} W_m^{J,k} K(\bar{x}_l^{I,k}, \bar{y}_m^{J,k}), \quad l = 1, \dots, n$$

4. Letting $f(\bar{x}_l^{I,0}) = g(\bar{x}_l^{I,0})$, for each subinterval I on level k, $1 \leq k \leq \kappa$, add the effect of the far-field sources by interpolating the field from the parent interval on level $k-1$ (L2L):

$$f(\bar{x}_l^{I,k}) = g(\bar{x}_l^{I,k}) + \sum_{\substack{\bar{x}_l^{J,k-1} \text{ in parent} \\ \text{interval of } I}} f(\bar{x}_l^{J,k-1}) S_n(\bar{x}_l^{I,k}, \bar{x}_l^{J,k-1}), \quad l = 1, \dots, n$$

5. Finally compute $f(x_i)$, where x_i is in subinterval I, by interpolating the far-field

approximation and adding the nearby interactions:

$$f(x_i) = \sum_{l=1}^{n} f(\bar{x}_l^{I,\kappa}) S_n(\bar{x}_l^{I,\kappa}, x_i) + \sum_{\substack{y_j \text{in nearest neighbor} \\ \text{intervals of } I}} K(x_i, y_j)\sigma_j, \quad i = 1, \ldots, N$$

An analogous algorithm can be written for the 3-D bbFMM using (6.4).

6.4 Fast Convolution Using SVD Compression

In the FMM the largest contribution to the computational cost is the multipole-to-local (M2L) operation described in step 3 of the bbFMM algorithm. As such the optimization of this operation is important for an efficient fast summation method. One way to reduce the cost is to produce a more compact multipole and local expansion. Here we propose using the singular value decomposition to compress the low-rank approximation generated by Chebyshev interpolation.

To find such a low-rank approximation of the kernel we look to minimize the approximation error with respect to a specified norm. For sums of the form (5.1) where the distribution of observation points and sources is assumed to be uniform, a natural estimate of the error introduced by replacing the kernel $K(x, y)$ with a low-rank approximation $\tilde{K}(x, y)$ is

$$\varepsilon = \left[\int_{-1}^{1} \int_{-1}^{1} [K(x, y) - \tilde{K}(x, y)]^2 \, dx dy \right]^{\frac{1}{2}}$$

where the domain of $K(x, y)$ is $[-1, 1] \times [-1, 1]$. This expression can be approximated using a Chebyshev quadrature for the double integral

$$\varepsilon' = \left[\sum_{l=1}^{n} \sum_{m=1}^{n} \omega_l^x \omega_m^y [K(\bar{x}_l, \bar{y}_m) - \tilde{K}(\bar{x}_l, \bar{y}_m)]^2 \right]^{\frac{1}{2}}$$

where $\{\bar{x}_l\}$ and $\{\bar{y}_m\}$ are the Chebyshev nodes and

$$\omega_l^x = \frac{\pi}{n}\sqrt{1-\bar{x}_l^2}$$
$$\omega_m^y = \frac{\pi}{n}\sqrt{1-\bar{y}_m^2}$$

are the corresponding weights. Defining the matrices $\mathbf{K}_{lm} = K(\bar{x}_l,\bar{y}_m)$, $\tilde{\mathbf{K}}_{lm} = \tilde{K}(\bar{x}_l,\bar{y}_m)$, $(\mathbf{\Omega^x})_{ll} = \omega_l^x$, and $(\mathbf{\Omega^y})_{mm} = \omega_m^y$, the error ε' can be expressed in terms of the Frobenius norm:

$$\varepsilon' = ||(\mathbf{\Omega^x})^{\frac{1}{2}}\mathbf{K}(\mathbf{\Omega^y})^{\frac{1}{2}} - (\mathbf{\Omega^x})^{\frac{1}{2}}\tilde{\mathbf{K}}(\mathbf{\Omega^y})^{\frac{1}{2}}||_F \tag{6.5}$$

Observe that for the low-rank approximation (6.3), we have $\tilde{\mathbf{K}} = \mathbf{K}$ which gives $\varepsilon' = 0$. However, we are interested in obtaining a compressed low-rank approximation, so we look for $\tilde{\mathbf{K}}$ such that: $\text{rank}(\tilde{\mathbf{K}}) < \text{rank}(\mathbf{K})$. The solution to this constrained minimization of (6.5) is given by a theorem from numerical linear algebra which states that the best rank-r approximation of an n-by-n matrix $\mathbf{A}$, where $r \leq n$, with respect to the Frobenius norm corresponds to picking the r left and right singular vectors of the SVD of $\mathbf{A}$ with the largest singular values [88]. Let the SVD of $(\mathbf{\Omega^x})^{\frac{1}{2}}\mathbf{K}(\mathbf{\Omega^y})^{\frac{1}{2}}$ be denoted by

$$(\mathbf{\Omega^x})^{\frac{1}{2}}\mathbf{K}(\mathbf{\Omega^y})^{\frac{1}{2}} = \mathbf{U\Sigma V}^T$$

where the columns of $\mathbf{U}$ are the left singular vectors, the columns of $\mathbf{V}$ are the right singular vectors, and $\mathbf{\Sigma}$ is a diagonal matrix whose entries are the singular values of $(\mathbf{\Omega^x})^{\frac{1}{2}}\mathbf{K}(\mathbf{\Omega^y})^{\frac{1}{2}}$ in order of decreasing magnitude. Since $\mathbf{\Omega^x}$ and $\mathbf{\Omega^y}$ are diagonal matrices of full rank, the optimal rank-r approximation of $\mathbf{K}$ is

$$\tilde{\mathbf{K}} = (\mathbf{\Omega^x})^{-\frac{1}{2}}\mathbf{U}_r\mathbf{\Sigma}_r\mathbf{V}_r^T(\mathbf{\Omega^y})^{-\frac{1}{2}} \tag{6.6}$$

where $\mathbf{U}_r$ is the first r columns of $\mathbf{U}$, $\mathbf{V}_r$ is the first r columns of $\mathbf{V}$, and $\mathbf{\Sigma}_r$ is a diagonal matrix containing the first r singular values. It should be noted that if the compression was done by computing the SVD of $\mathbf{K}$ instead of $(\mathbf{\Omega^x})^{\frac{1}{2}}\mathbf{K}(\mathbf{\Omega^y})^{\frac{1}{2}}$

then, using a similar argument as above, it can be shown that the error in the low-rank approximation is minimized for a distribution of sources and observation points concentrated on the boundaries. However for a uniform distribution, the reduced-rank matrix in (6.6) gives the optimal compression.

We now proceed to show how to compress the M2L operator using the SVD compression described above. With the same notation as in the previous section, the M2L operation between observation points $\{\bar{x}_l\}$ and sources located at $\{\bar{y}_m\}$ can be expressed as the matrix-vector product

$$\mathbf{g} = \mathbf{K}\mathbf{w}$$

where $\mathbf{g}_l = g(\bar{x}_l)$, $\mathbf{K}_{lm} = K(\bar{x}_l, \bar{y}_m)$, and $\mathbf{w}_m = W_m$. Then $\mathbf{g}$ can be approximated by

$$\tilde{\mathbf{g}} = \tilde{\mathbf{K}}\mathbf{w} = (\mathbf{\Omega^x})^{-\frac{1}{2}}\mathbf{U}_r\mathbf{\Sigma}_r\mathbf{V}_r^T(\mathbf{\Omega^y})^{-\frac{1}{2}}\mathbf{w}.$$

The compressed low-rank approximation reduces the cost from an n-by-n matrix-vector product to two n-by-r matrix-vector products. This calculation can be further streamlined so as to involve mainly r-by-r matrices. To begin consider the M2L operation in 3-D for an observation cell on level k of the FMM tree. Each observation cell interacts with up to $6^3 - 3^3 = 189$ source cells, with each interaction indexed by its transfer vector. The union of transfer vectors over all observation cells on level k forms a set of $7^3 - 3^3 = 316$ vectors. **Assuming a translational invariant kernel,** there are 316 unique M2L operators on level k, each one corresponding to a particular transfer vector. Let $\mathbf{K}^{(i)}$ denote the 3-D M2L operator for the i-th transfer vector. Then this collection of M2L operators, with the appropriate weighting in 3-D, can be expressed either as a fat matrix

$$\mathbf{K}_{\text{fat}} = [(\mathbf{\Omega^x})^{\frac{1}{2}}\mathbf{K}^{(1)}(\mathbf{\Omega^y})^{\frac{1}{2}} \quad (\mathbf{\Omega^x})^{\frac{1}{2}}\mathbf{K}^{(2)}(\mathbf{\Omega^y})^{\frac{1}{2}} \quad \cdots \quad (\mathbf{\Omega^x})^{\frac{1}{2}}\mathbf{K}^{(316)}(\mathbf{\Omega^y})^{\frac{1}{2}}]$$

with the $(\mathbf{\Omega^x})^{\frac{1}{2}}\mathbf{K}^{(i)}(\mathbf{\Omega^y})^{\frac{1}{2}}$ blocks arranged in a single row, or as a thin matrix

$$\mathbf{K}_{\text{thin}} = [(\mathbf{\Omega^x})^{\frac{1}{2}}\mathbf{K}^{(1)}(\mathbf{\Omega^y})^{\frac{1}{2}}; \quad (\mathbf{\Omega^x})^{\frac{1}{2}}\mathbf{K}^{(2)}(\mathbf{\Omega^y})^{\frac{1}{2}}; \quad \cdots; \quad (\mathbf{\Omega^x})^{\frac{1}{2}}\mathbf{K}^{(316)}(\mathbf{\Omega^y})^{\frac{1}{2}}]$$

with the $(\mathbf{\Omega}^{\mathbf{x}})^{\frac{1}{2}}\mathbf{K}^{(i)}(\mathbf{\Omega}^{\mathbf{y}})^{\frac{1}{2}}$ blocks arranged in a single column. Here $\mathbf{\Omega}^{\mathbf{x}}$ and $\mathbf{\Omega}^{\mathbf{y}}$ are the 3-D analogs of the weighting matrices used in (6.5).

To construct compact multipole and local expansions we perform two SVDs, one on $\mathbf{K}_{\text{fat}}$ and the other on $\mathbf{K}_{\text{thin}}$:

$$\begin{aligned}
\mathbf{K}_{\text{fat}} &= [(\mathbf{\Omega}^{\mathbf{x}})^{\frac{1}{2}}\mathbf{K}^{(1)}(\mathbf{\Omega}^{\mathbf{y}})^{\frac{1}{2}} \quad (\mathbf{\Omega}^{\mathbf{x}})^{\frac{1}{2}}\mathbf{K}^{(2)}(\mathbf{\Omega}^{\mathbf{y}})^{\frac{1}{2}} \quad \cdots \quad (\mathbf{\Omega}^{\mathbf{x}})^{\frac{1}{2}}\mathbf{K}^{(316)}(\mathbf{\Omega}^{\mathbf{y}})^{\frac{1}{2}}] \\
&= \mathbf{U}\mathbf{\Sigma}[\mathbf{V}^{(1)^T}\mathbf{V}^{(2)^T}\cdots\mathbf{V}^{(316)^T}] \\
\mathbf{K}_{\text{thin}} &= [(\mathbf{\Omega}^{\mathbf{x}})^{\frac{1}{2}}\mathbf{K}^{(1)}(\mathbf{\Omega}^{\mathbf{y}})^{\frac{1}{2}}; \quad (\mathbf{\Omega}^{\mathbf{x}})^{\frac{1}{2}}\mathbf{K}^{(2)}(\mathbf{\Omega}^{\mathbf{y}})^{\frac{1}{2}}; \quad \cdots; \quad (\mathbf{\Omega}^{\mathbf{x}})^{\frac{1}{2}}\mathbf{K}^{(316)}(\mathbf{\Omega}^{\mathbf{y}})^{\frac{1}{2}}] \\
&= [\mathbf{R}^{(1)};\mathbf{R}^{(2)};\cdots;\mathbf{R}^{(316)}]\mathbf{\Lambda}\mathbf{S}^T.
\end{aligned}$$

Observe that if the kernel is symmetric then $\mathbf{K}_{\text{thin}} = \mathbf{K}_{\text{fat}}^T$ so the two SVDs are just transposes of each other. The pre-computation cost for these SVDs is $O(\kappa)$ since the dimensions of these matrices are independent of the problem size.

The cost of the convolution for the i-th transfer vector between $\mathbf{K}^{(i)}$ and a vector of sources $\mathbf{w}$ can be reduced by employing these two SVDs as follows. First begin by introducing the weighting matrices $\mathbf{\Omega}^{\mathbf{x}}$ and $\mathbf{\Omega}^{\mathbf{y}}$ and substituting in the i-th block of $\mathbf{K}_{\text{thin}}$.

$$\begin{aligned}
\mathbf{K}^{(i)}\mathbf{w} &= (\mathbf{\Omega}^{\mathbf{x}})^{-\frac{1}{2}}(\mathbf{\Omega}^{\mathbf{x}})^{\frac{1}{2}}\mathbf{K}^{(i)}(\mathbf{\Omega}^{\mathbf{y}})^{\frac{1}{2}}(\mathbf{\Omega}^{\mathbf{y}})^{-\frac{1}{2}}\mathbf{w} \\
&= (\mathbf{\Omega}^{\mathbf{x}})^{-\frac{1}{2}}\mathbf{R}^{(i)}\mathbf{\Lambda}\mathbf{S}^T(\mathbf{\Omega}^{\mathbf{y}})^{-\frac{1}{2}}\mathbf{w}
\end{aligned}$$

Next the identity matrix $\mathbf{S}^T\mathbf{S}$ can be inserted between $\mathbf{\Lambda}$ and $\mathbf{S}^T$ after which the i-th block of $\mathbf{K}_{\text{thin}}$ is replaced.

$$\begin{aligned}
\mathbf{K}^{(i)}\mathbf{w} &= (\mathbf{\Omega}^{\mathbf{x}})^{-\frac{1}{2}}\mathbf{R}^{(i)}\mathbf{\Lambda}\mathbf{S}^T\mathbf{S}\mathbf{S}^T(\mathbf{\Omega}^{\mathbf{y}})^{-\frac{1}{2}}\mathbf{w} \\
&= (\mathbf{\Omega}^{\mathbf{x}})^{-\frac{1}{2}}(\mathbf{\Omega}^{\mathbf{x}})^{\frac{1}{2}}\mathbf{K}^{(i)}(\mathbf{\Omega}^{\mathbf{y}})^{\frac{1}{2}}\mathbf{S}\mathbf{S}^T(\mathbf{\Omega}^{\mathbf{y}})^{-\frac{1}{2}}\mathbf{w}
\end{aligned}$$

Now substituting in the i-th block of $\mathbf{K}_{\text{fat}}$ and inserting the identity matrix $\mathbf{U}^T\mathbf{U}$

between $\mathbf{U}$ and $\mathbf{\Sigma}$ we have

$$\begin{aligned}\mathbf{K}^{(i)}\mathbf{w} &= (\mathbf{\Omega}^{\mathbf{x}})^{-\frac{1}{2}}\mathbf{U}\mathbf{\Sigma}\mathbf{V}^{(\mathbf{i})^T}\mathbf{S}\mathbf{S}^T(\mathbf{\Omega}^{\mathbf{y}})^{-\frac{1}{2}}\mathbf{w} \\ &= (\mathbf{\Omega}^{\mathbf{x}})^{-\frac{1}{2}}\mathbf{U}\mathbf{U}^T\mathbf{U}\mathbf{\Sigma}\mathbf{V}^{(\mathbf{i})^T}\mathbf{S}\mathbf{S}^T(\mathbf{\Omega}^{\mathbf{y}})^{-\frac{1}{2}}\mathbf{w} \\ &= (\mathbf{\Omega}^{\mathbf{x}})^{-\frac{1}{2}}\mathbf{U}\left[\mathbf{U}^T(\mathbf{\Omega}^{\mathbf{x}})^{\frac{1}{2}}\mathbf{K}^{(i)}(\mathbf{\Omega}^{\mathbf{y}})^{\frac{1}{2}}\mathbf{S}\right]\mathbf{S}^T(\mathbf{\Omega}^{\mathbf{y}})^{-\frac{1}{2}}\mathbf{w}.\end{aligned}$$

Consider the term inside the square brackets:

$$\mathbf{U}^T(\mathbf{\Omega}^{\mathbf{x}})^{\frac{1}{2}}\mathbf{K}^{(i)}(\mathbf{\Omega}^{\mathbf{y}})^{\frac{1}{2}}\mathbf{S} = \mathbf{\Sigma}\mathbf{V}^{(i)^T}\mathbf{S} = \mathbf{U}^T\mathbf{R}^{(i)}\mathbf{\Lambda}.$$

This shows that the rows and columns of $\mathbf{U}^T(\mathbf{\Omega}^{\mathbf{x}})^{\frac{1}{2}}\mathbf{K}^{(i)}(\mathbf{\Omega}^{\mathbf{y}})^{\frac{1}{2}}\mathbf{S}$ decay as quickly as the singular values found in $\mathbf{\Sigma}$ and $\mathbf{\Lambda}$. Hence the product $\mathbf{K}^{(i)}\mathbf{w}$ can be approximated by keeping only the first r singular vectors in each of the SVDs. Let $\mathbf{U}_r$ and $\mathbf{S}_r$ denote the r left singular vectors and r right singular vectors respectively. Using these reduced SVDs a fast convolution method involving compressed multipole and local expansions can be formulated. The M2L operation, step 3 in the bbFMM algorithm, can now be done as follows. Using the notation adopted in the bbFMM algorithm we have:

0. Pre-computation: compute the compressed M2L operators for all transfer vectors $i = 1, \ldots, 316$ on level k, $0 \le k \le \kappa$

$$\mathbf{C}^{(i),k} = (\mathbf{U}_r^k)^T(\mathbf{\Omega}^{\mathbf{x}})^{\frac{1}{2}}\mathbf{K}^{(i),k}(\mathbf{\Omega}^{\mathbf{y}})^{\frac{1}{2}}\mathbf{S}_r^k$$

3a. Pre-processing: compute the compressed multipole coefficients for all source cells I on level k, $0 \le k \le \kappa$

$$(\mathbf{w_c})^{I,k} = (\mathbf{S}_r^k)^T(\mathbf{\Omega}^{\mathbf{y}})^{-\frac{1}{2}}\mathbf{w}^{I,k}$$

3b. Convolution: calculate the compressed local coefficients for all observation cells I on level k, $0 \le k \le \kappa$; let $i(I, J)$ denote the index of the transfer vector representing the interaction between cells I and J

$$(\mathbf{g_c})^{I,k} = \sum_{\substack{\text{cells } J \text{ in interaction} \\ \text{list of cell } I}} \mathbf{C}^{(i(I,J)),k}(\mathbf{w_c})^{J,k}$$

3c. Post-processing: compute the coefficients of the local expansion for all observation cells I on level k, $0 \leq k \leq \kappa$

$$\mathbf{g}^{I,k} = (\mathbf{\Omega}^{\mathbf{x}})^{-\frac{1}{2}} \mathbf{U}_r^k (\mathbf{g_c})^{I,k}$$

Most of the computational cost in the fast convolution algorithm is concentrated in step 3b which involves **reduced-rank** matrix-vector products. Without the SVD compression, the cost of the M2L operation corresponds to **full-rank** matrix-vector products.

The cost and memory requirements of the pre-computation step can be reduced for the case of homogeneous kernels. Recall that a function $K(\mathbf{x}, \mathbf{y})$ is homogeneous of degree m if $K(\alpha\mathbf{x}, \alpha\mathbf{y}) = \alpha^m K(\mathbf{x}, \mathbf{y})$ for any nonzero real α. In this case the M2L operators can be determined for interactions between observation and source cubes with unit volume and two SVDs are performed. Letting $\mathbf{D}^{(i)}$ represent the compressed M2L operators constructed from these two SVDS then, for a cubic computational cell with sides of length L, the operators on each level of the FMM tree are scaled versions of $\mathbf{D}^{(i)}$:

$$\mathbf{C}^{(i),k} = \left(\frac{L}{2^k}\right)^m \mathbf{D}^{(i)}.$$

Hence only one set of operators, $\{\mathbf{D}^{(i)}\}$, needs to computed and stored because $\mathbf{C}^{(i),k}$ can be easily computed from $\mathbf{D}^{(i)}$. In addition only the singular vectors from the two SVDs are needed for the fast convolution algorithm because singular vectors are invariant under multiplicative scaling of the matrix. In that case the cost of the pre-computation is $O(1)$ for any problem size.

This is summarized in Table I. In the general case, the SVD provides only limited savings because a new SVD has to be computed for every cluster. If the method is applied many times for a given tree, this is still computationally advantageous.

Kernel type	Cost
General case	$O(N)$
Translational invariant	$O(\kappa)$
Homogeneous	$O(1)$
Symmetric	Cost is reduced by 2

Table I: Computational cost of pre-computation. The pre-computation includes all steps which depend only on the boxes in the tree and are independent of the particles' location and σ_j. The kernel is $K(x, y)$. Notations: N: number of particles, κ: number of levels.

6.5 Numerical Results

In this section we will present numerical results for the bbFMM. The accuracy of the method will be examined as well as the computational cost. Results for four kernels will be detailed: (1) the Laplacian kernel $1/r$, (2) the kernel $1/r^4$, (3) the Stokes kernel $I_{3\times3}/r + (\vec{r} \otimes \vec{r})/r^3$ where $\vec{r}$ is the 3-dimensional position vector, and (4) the 3-D multiquadric radial basis function $\sqrt{(r/a)^2 + 1}$ where a is a scaling constant.

6.5.1 Compression using SVD

We start by examining the amount of compression that can be achieved in the M2L operation by using the SVDs of the kernel matrices $\mathbf{K}_{\text{fat}}$ and $\mathbf{K}_{\text{thin}}$. In Fig. 6.1 the singular values of the Laplacian kernel matrices are plotted for various number of Chebyshev nodes n. Since the Laplacian kernel is symmetric the singular values of $\mathbf{K}_{\text{fat}}$ and $\mathbf{K}_{\text{thin}}$ are identical. For each n the singular values are scaled such that the largest singular value is normalized to 1. The index of the singular values $(1, \dots, n^3)$ is represented on the horizontal axis. The left subfigure shows the singular values obtained by setting up the kernel matrices and performing the SVD in double precision while the right subfigure corresponds to single precision. Taking the curve for $n = 10$ as the best approximation to the continuous SVD, observe that the double-precision singular values are accurate up to approximately the index $n^3/2$ [26]. This suggests that the kernel matrices can be compressed by a factor of 2 without adversely affecting the accuracy of the method. In the single-precision plot, we see that the amount of compression is less as the curves deviate at an index greater than $n^3/2$. The leveling of the curves around 10^{-8} reflects the roundoff error incurred by using single precision.

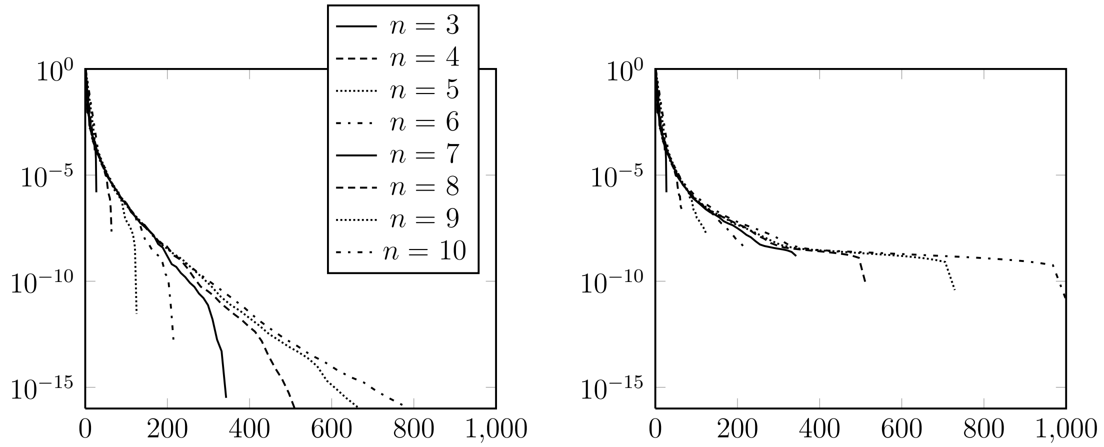
Figs. 6.2 and 6.3 are similar plots for the $1/r^4$ and Stokes kernels, respectively. For the Stokes kernel the indices of the singular values are $1, \dots, 3n^3$.

While the rate of decay of the singular values for homogeneous kernels is scale-invariant, this is not the case for inhomogeneous kernels such as the 3-D multiquadric radial basis functions $\sqrt{(r/a)^2 + 1}$. In Fig. 6.4 the double-precision singular values for two radial basis functions, $a = 1$ and $a = 8$, are shown. Observe that when $a \ll 1$ or $a = O(1)$ (e.g., Fig. 6.4(a)) the profile is similar to that obtained for the Laplacian kernel and hence the kernel matrices can be compressed by keeping only half of the singular values. However when $a \gg 1$ (e.g., Fig. 6.4(b)) the decay is much more rapid since the radial basis function is well-approximated by the constant 1. The implication of this behavior for the multilevel FMM scheme is that fewer singular values can be retained for deeper levels in the tree, thereby reducing the computational cost. This illustrates that in order to achieve the best compression for a particular kernel, the decay behavior of the singular values needs to be studied on a case-by-case basis.

In all subsequent benchmarks, the Laplacian kernel matrices were compressed by retaining only the largest $n^3/2$ singular values.

6.5.2 Interpolation Error

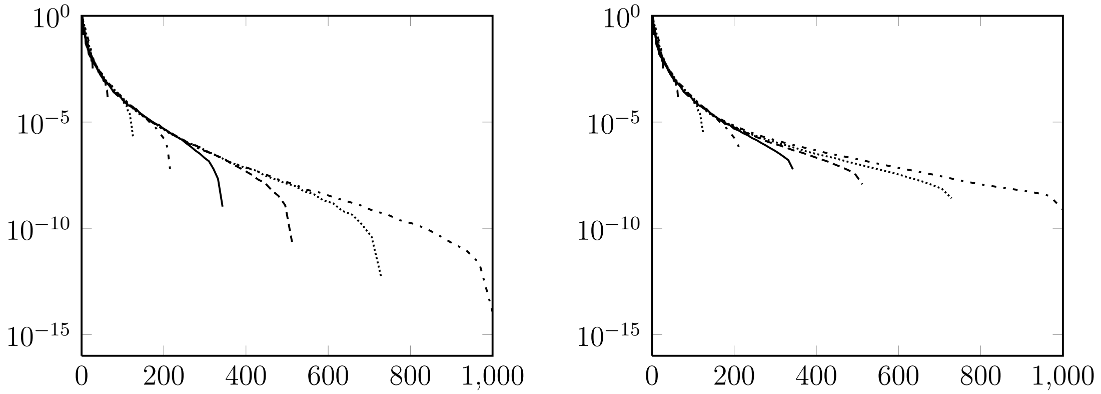
To investigate the Chebyshev interpolation error we looked at a system of 10,000 uniformly distributed charges in a cubic computational domain with edge length 1. Each charge was assigned a strength of $+1$ or -1 such that the computational cell has zero net charge. The observation points were chosen to be identical to the charge locations and the charges interacted according to the Laplacian kernel $1/r$. To compute the error only, a subset of 100 observation points was used. The relative error in the observation values was measured with respect to the 2-norm and the inf-norm for various n by using the values obtained by direct calculation as the reference solution. Letting $f^{\mathrm{FMM}}(x_i)$ and $f^{\mathrm{dir}}(x_i)$ be the observation values computed with



(a) Double Precision

(b) Single Precision

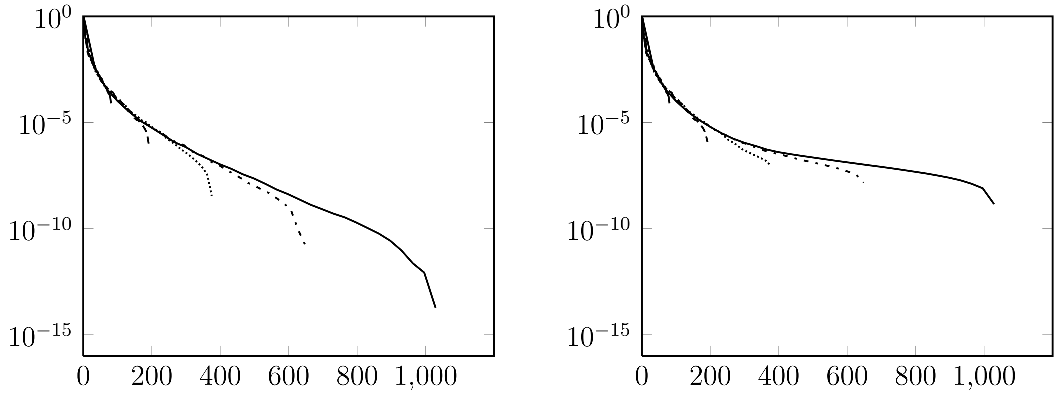
Figure 6.1: Singular values for the Laplacian kernel. The relative singular value magnitude (vertical axis) is plotted as a function of the singular value index (shown on the horizontal axis). The subsequent plots (Figs. 6.2-6.4) show the decay of the singular values for three other kernels. The legend is the same for all plots.



(a) Double Precision

(b) Single Precision

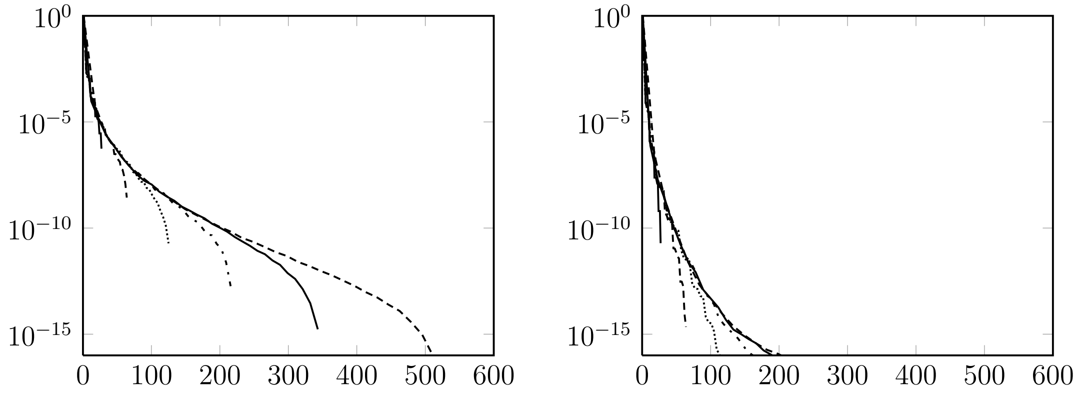
Figure 6.2: Singular values for the $1/r^4$ kernel



(a) Double Precision

(b) Single Precision

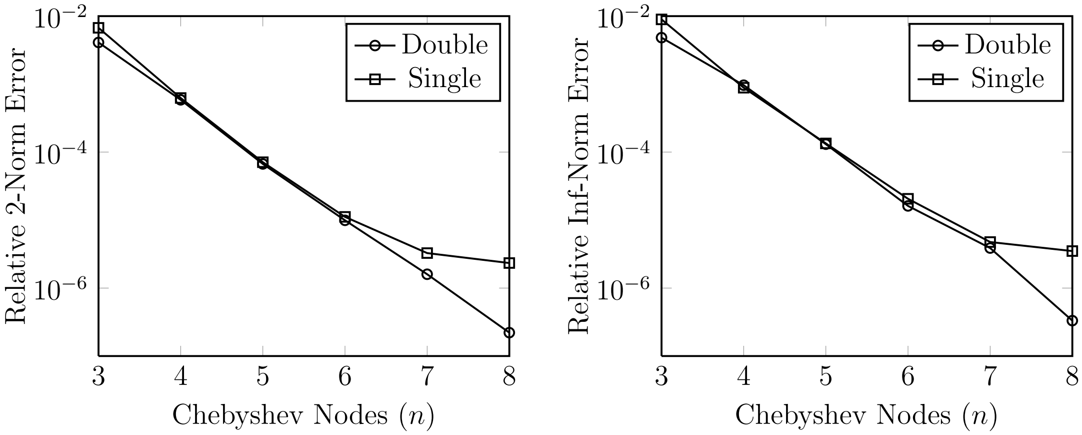
Figure 6.3: Singular values for the Stokes kernel. In this plot, n ranges from 3 to 7.



(a) $\sqrt{r^2+1}$

(b) $\sqrt{(r/8)^2+1}$

Figure 6.4: Singular values for 3-D multiquadric radial basis function. In this plot, n ranges from 3 to 8. Double precision was used.



(a) Relative 2-Norm

(b) Relative Inf-Norm

Figure 6.5: Interpolation error for the Laplacian kernel. The relative 2-norm and inf-norm errors are defined by Eqs. (6.7) and (6.8) respectively.

bbFMM and direct calculation, respectively, the errors are given by

$$\varepsilon_2 = \left[\frac{\sum_{i=1}^{100} \left(f^{\mathrm{FMM}}(x_i) - f^{\mathrm{dir}}(x_i) \right)^2}{\sum_{i=1}^{100} \left(f^{\mathrm{dir}}(x_i) \right)^2} \right]^{1/2} \tag{6.7}$$

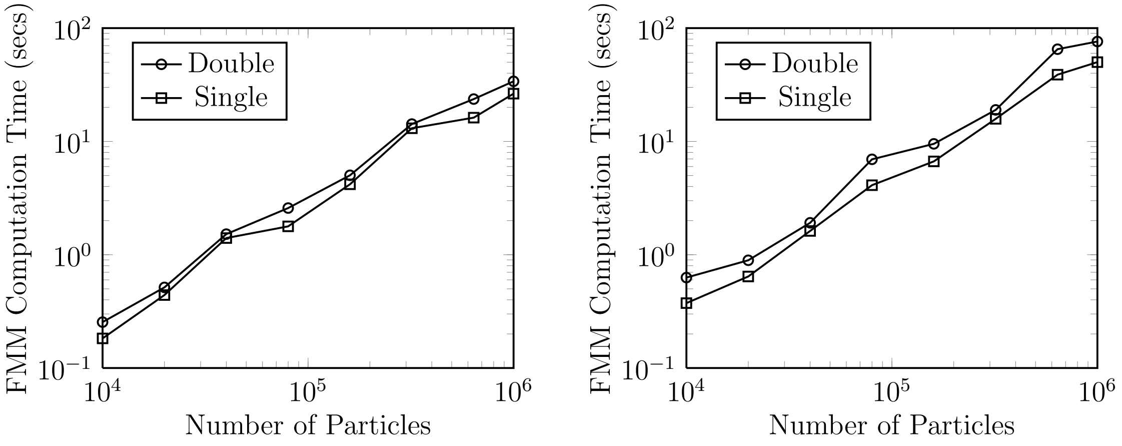
and

$$\varepsilon_\infty = \frac{\max_{1 \le i \le 100} \left| f^{\mathrm{FMM}}(x_i) - f^{\mathrm{dir}}(x_i) \right|}{\max_{1 \le i \le 100} \left| f^{\mathrm{dir}}(x_i) \right|}. \tag{6.8}$$

Fig. 6.5 shows that for the Laplacian kernel the error in both norms exhibits spectral convergence when double precision is used. However for single precision the error levels off for large n as roundoff error degrades the solution.

6.5.3 Computational Cost

To examine the computational complexity of bbFMM, a system of N uniformly distributed charges in a cubic computational domain with edge length 1 was used. Each charge was assigned a strength of $+1$ or -1 such that the computational cell has zero net charge. The observation points were chosen to be identical to the charge



(a) $n = 4$ (b) $n = 5$

Figure 6.6: Computational cost for the Laplacian kernel. The number of Chebyshev nodes n was varied for the 2 plots.

locations and the charges interacted according to the Laplacian kernel $1/r$. As the number of charges N was varied from 10^4 to 10^6 the FMM computation time (cost of the M2M, M2L, and L2L operations and cost of direct interactions) was measured for both double and single precision calculations and plotted in Fig. 6.6. The number of Chebyshev nodes was chosen to be either $n = 4$ or $n = 5$. The number of levels in the FMM tree was selected to maintain roughly a constant number of charges per cell on the finest level, i.e., $\kappa = O(\ln N)$. In both instances the correct $O(N)$ complexity is observed. The kinks in the curves result from increases in the number of levels in the tree.

6.5.4 Comparison with Analytic Multipole Expansion

To study the efficiency of the SVD compression, a comparison was done with the analytic multipole expansion of the Laplacian kernel using Legendre polynomials for a system consisting of two well-separated cubes. The source cube was centered at the origin while the observation cube was centered at $(1/2, 0, 0)$. Both cubes have edge length $1/4$. Letting $\vec{r}_i$ denote the position vector of the i-th observation point and $\vec{r}_j$ the position of the j-th source, the p-th order analytic multipole expansion of the observational value is given by

$$f(\vec{r}_i) = \frac{1}{r_i} \sum_{j=1}^{20} q_j \sum_{l=0}^{p-1} \left(\frac{r_j}{r_i} \right)^l P_l(\cos\theta_{ij}) \tag{6.9}$$

where P_l is the Legendre polynomial of degree l and

$$\cos\theta_{ij} = \frac{\vec{r}_i^{\,T} \vec{r}_j}{r_i \, r_j}$$

Since P_l can be replaced by $2l+1$ spherical harmonics, this expansion can be rewritten exactly in terms of a finite number of spherical harmonics, which are used in the far field expansion. The number of spherical harmonics coefficients for a p-th order multipole expansion is

$$\sum_{l=0}^{p-1} (2l+1) = p + 2 \sum_{l=0}^{p-1} l = p + (p-1)p = p^2.$$

This comparison was carried out for two systems. The first system is constructed by placing $6^3 - 4^3 = 152$ charges on the faces of the cubes, such that, on each face, we have $6^2 = 36$ charges distributed on a regular grid. Charge strengths were alternated between $+1$ and -1 in a checkerboard fashion. In the second system the 152 charges were uniformly distributed within each cube. For various values of p, the 2-norm error in the observational values was computed using the values obtained by direct calculation as the reference solution. The error was also determined when retaining p^2 singular values in the bbFMM. We used $n = 10$ Chebyshev nodes in each direction, for all the tests. Fig. 6.7 shows the errors for the two systems. The values of p^2, which were varied from $1^2 = 1$ to $16^2 = 256$, are plotted on the horizontal axis. From the plots, the SVD compression is at least as good as the analytic multipole expansion with respect to the 2-norm error. For the case on the left (charges confined to the surface) the difference between the two methods is more substantial for large p. This is because the multipole expansion is similar to a Taylor series and therefore does not do a good job of approximating charges near the boundaries of the computational

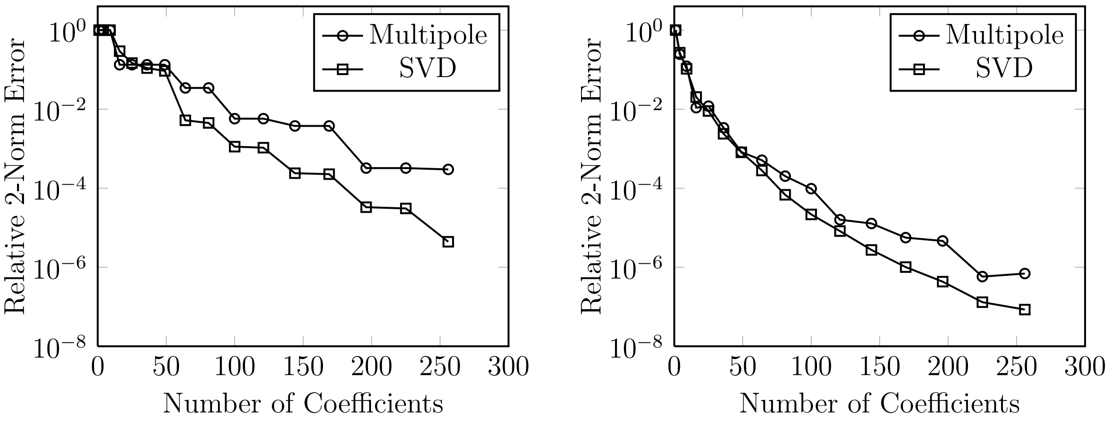


Figure 6.7: Error comparison between the SVD compression and the analytic multipole expansion. Left: charges distributed on the surface of the cubes. Right: random uniform charge distribution.

domain. The SVD compression with the Chebyshev-based expansion, however, is able to resolve those charges more accurately.

6.6 Summary

We have presented a new black-box or kernel-independent fast multipole method. The method requires as input only a user defined routine to numerically evaluate $K(x, y)$ at a given point (x, y). This is very convenient for complex kernels, for which analytical expansions might be difficult to obtain. This method relies on Chebyshev polynomials for the interpolation part and on singular value decompositions to further reduce the computational cost. Because of the SVD, we can prove that the scheme uses the minimal number of coefficients given a tolerance ε. The pre-computing time of the method was analyzed and was found to be small for most practical cases. The numerical scheme was tested on various problems. The accuracy was confirmed and spectral convergence was observed. The linear complexity was also confirmed by numerical experiments.

A limitation of the current approach is that the M2L operator is dense. This, probably, cannot be avoided if one uses an SVD. However, if one agrees to choose a different expansion, involving more terms, it is sometimes possible to design methods with diagonal M2L operators. Even though more coefficients are used, one may end up with a faster algorithm overall. This is the case for example with the plane wave version of the FMM for $1/r$ [37]. In that case, a specialized algorithm can be faster than the proposed scheme. It remains to be seen if a black box method can be derived using diagonal operators.

Another issue is the extension to periodic boundary conditions, in particular to the case of conditionally convergent series like $K(x, y) = 1/r$, which converge only for a cell with zero net charge. This topic will be explored further in the next chapter.

Chapter 7

A Fast Multipole Method for Periodic Systems

So far we have focused on computing pairwise interactions for strictly non-periodic systems. However there are applications where it is desirable to impose periodic boundary conditions. For instance the simulation of a small periodic system is a practical approach for studying bulk material properties. In this chapter we will extend the black-box fast multipole method (bbFMM) discussed in the previous chapter to periodic systems. To begin we describe how bbFMM can be used to compute absolutely convergent periodic sums. Next we discuss why conditionally convergent sums will not converge when bbFMM is applied directly. Finally we conclude by describing how bbFMM can be modified to produce a convergent result that is periodic.

7.1 Extending bbFMM for Absolutely Convergent Periodic Sums

Consider the 3-D periodic sum

$$f(\mathbf{x}_i) = \sum_{\mathbf{n}} \sum_{j=1}^{N} K(\mathbf{x}_i, \mathbf{y}_j + \mathbf{n}L)\sigma_j, \quad i = 1, \ldots, N \tag{7.1}$$

where $\mathbf{n} = (n_1, n_2, n_3)$ is a vector of integers indexing the periodic images of a cubic computational cell with edge length L. If the kernel $K(\mathbf{x}, \mathbf{y})$ scales like $1/r^p$ where $p \geq 3+\delta$ for some positive constant δ, then this periodic sum is absolutely convergent [add citation]. In this instance the sum converges to the same value regardless of the order of summation or the limit of truncation. Hence one approach for calculating the sum is to simply extend the FMM tree discussed in Sec. 5.2 to include the periodic images.

Using the algorithm outlined in Sec. 6.3 the interactions due to all sources located in the original computational domain and its 26 neighboring cells can be computed. This is accomplished by treating these 27 cells as near neighbors of the original cell at level 0 of the FMM tree. To calculate the contributions from the periodic images with $|n_i| > 1$ for some i, levels with negative indices are added to the FMM tree. On level k where $k \leq 0$, the edge length of the computational cubes are $3^{-k}L$. Note this choice is equivalent to subdividing a parent cell into 27 children, which differs from the octree structure for levels indexed by positive k (see Sec. 5.2). Geometrically the periodic sum is computed using a series of spherical shells whose radii are exponentially increasing.

Computation on the levels of the FMM tree corresponding to the periodic images involves M2M and M2L operations analogous to those given in Sec. 6.3. Since the weights at the Chebyshev nodes are identical for all periodic images, the M2M and M2L operations can be simply expressed as matrix-vector products where the matrices have dimension $n^3 \times n^3$. In addition these matrix operators can be pre-computed because the location of the Chebyshev nodes are fixed. It should be noted that no L2L operation is needed because we are only interested in the local expansion in the original computational cell so all expansions computed by the M2L operation will be with respect to this cell. Therefore the contribution to the local expansion on level 0 of the FMM tree from the non-neighboring periodic images can written compactly as

$$g(\bar{\mathbf{x}}^{I,0}) = \sum_{k=-\infty}^{0} \mathbf{K}^{(k)} \ (\mathbf{S}_{\mathrm{p}})^{|k|} \ \mathbf{W}^{I,0} \tag{7.2}$$

where $\mathbf{K}^{(k)}$ is the M2L operator on level k, $\mathbf{S}_\mathrm{p}$ is the M2M operator, $\bar{\mathbf{x}}^{I,0}$ is the vector of Chebyshev node locations in the computational domain, and $\mathbf{W}^{I,0}$ are corresponding weights at the nodes. Since the M2M operator is identical for all levels, the weights at level k can be obtained by simply applying $\mathbf{S}_\mathrm{p}$ k times.

7.2 Handling Conditionally Convergent Periodic Sums

For three-dimensional kernels $K(\mathbf{x}, \mathbf{y})$ that scale like $1/r^p$ where $p \leq 3$, the sum (7.1) is conditionally convergent when the cell has zero net charge. As such the summation over the periodic images must be handled carefully. To determine whether (7.2) can be used for conditionally convergent sums, we will look at the matrices $\mathbf{K}^{(k)}$ and $\mathbf{S}_\mathrm{p}$ more closely. Assuming that the kernel scales like $1/r^p$ then the entries in $\mathbf{K}^{(k)}$ are roughly proportional to $1/(3^{|k|}L)^p$ where $3^{|k|}L$ is the characteristic length on level k. For the M2M matrix $\mathbf{S}_\mathrm{p}$ the largest eigenvalue is 27, which corresponds to the conservation of net charge from the 27 child cells to the parent cell. Therefore we have

$$g(\bar{\mathbf{x}}^{I,0}) = \sum_{k=-\infty}^{0} \mathbf{K}^{(k)}\ \mathbf{S}_\mathrm{p}^{|k|}\ \mathbf{W}^{I,0} \propto \sum_{k=-\infty}^{0} \left(\frac{1}{3^{|k|}L}\right)^p (3^3)^{|k|} \propto \sum_{k=-\infty}^{0} \left(3^{3-p}\right)^{|k|}$$

thus Eq. (7.2) is proportional to an infinite geometric series. From this analysis we see that the series converges only if $p \geq 3 + \delta$ for some positive constant δ. Hence (7.2) cannot be used directly when $p \leq 3$.

One approach for treating conditionally convergent periodic sums is detailed in the reference [add citation]. Here the conditional convergence of elastic fields of dislocations is handled by observing that the periodic sum can be made absolutely convergent by taking spatial derivatives of the sum. For example only one derivative is needed for kernels proportional to $1/r^3$ while two derivatives are needed for kernels that scale like $1/r^2$. The original periodic sum is recovered by integration in space, which introduces a constant field for $p = 3$ while for $p = 2$ linear and constant fields need to be added. To determine these unknown integration constants the sum (7.1) is enforced to be periodic. For example to find the coefficients of the linear field, we can enforce the condition that $f(\mathbf{x})$ should be equal at the corners of the computational cube since under periodic boundary conditions they all represent the same point. The constant field can be computed by using additional information e.g. using that the kernel is the gradient of a potential function.

Extending these ideas to bbFMM, we would like to modify the kernel $K(\mathbf{x}, \mathbf{y})$ such that we have an absolutely convergent sum. Take as an example the electrostatic force kernel

$$K(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x} - \mathbf{y}}{|\mathbf{x} - \mathbf{y}|^3}, \tag{7.3}$$

a kernel that decays like $1/r^2$. Noting that the sum (7.1) is conditionally convergent for a cell with zero net charge, an absolutely convergent kernel can be obtained by introducing an additional term into the periodic sum:

$$f(\mathbf{x}_i) = \sum_{\mathbf{n}} \sum_{j=1}^{N} \left[K(\mathbf{x}_i, \mathbf{y}_j + \mathbf{n}L) - K(\mathbf{x}_i, \mathbf{n}L) \right] \sigma_j. \tag{7.4}$$

Observe that the contribution from the $K(\mathbf{x}_i, \mathbf{n}L)$ term is zero since the net charge in the cell is identically zero. The motivation behind this formulation is that in the far field, the interactions are point-dipole instead of point-point. By writing the periodic sum in the form of (7.4) this observation is enforced explicitly since

$$K(\mathbf{x}, \mathbf{y} + \mathbf{y}_j) - K(\mathbf{x}, \mathbf{y}) \sim \nabla K(\mathbf{x}, \mathbf{y}) \mathbf{y}_j.$$

If the kernel K is taken to be the difference inside the brackets of (7.4) then we now have a $O(1/r^3)$ kernel. However the the symmetry involved in the spherical shell summation of these point-dipole interactions results in an absolutely convergent sum.

7.3 Numerical Example

To illustrate how bbFMM can be used to compute conditionally convergent sums we will look at the electrostatic force kernel (7.3) for a system with two opposite charges in a unit cube centered at the origin under periodic boundary conditions. The electrostatic force for each charge is then calculated with bbFMM and compared to the reference value obtained by Ewald summation. Using Eq. (7.4) we evaluate the periodic sum along the line given by the parametric equations $(x, y, z) = (-0.5+t, 0, 0.5)$ which lies on a face of the computational cube domain. In Fig. 7.1 the x-component of the electrostatic force is plotted along this line and the boundary behavior is examined. Since the system is periodic we expect the force to be continuous across the edge of the cube at $x = -0.5$. The non-periodicity is due to the missing linear field correction. The unknown coefficients of the linear field $ax+by+cz$ can be determined by enforcing that the force is equal at the eight corners of the computational cube. After correcting for the non-periodic linear field, the force is now periodic along the specified line.

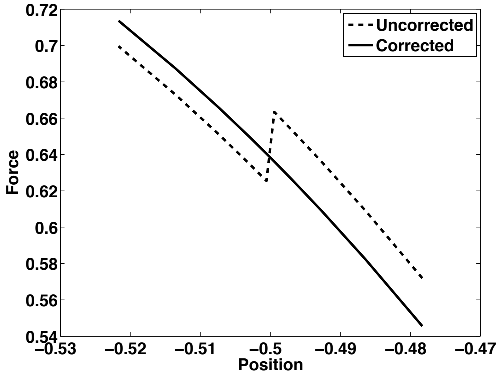


Figure 7.1: The periodic sum with the electrostatic force kernel is evaluated along the line $(x, y, z) = (-0.5+t, 0, 0.5)$ and the x-component of the force is plotted along this line. Since the system is periodic, the force is expected to be continuous across the edge located at $x = 0.5$. Without the linear field correction (dashed line) there is a jump in the force. After adding the non-periodic linear field, found by enforcing that the force is equal at the eight corners of the cube, the force is now continuous (solid line).

Charge	bbFMM	Ewald
1	-37.23955	-37.23962
2	37.23960	37.23962

Table I: Computation of the electrostatic force for a periodic system containing two opposite charges in a unit cube. The force in the x-direction for each charge was calculated using bbFMM and Ewald summation.

To calculate the constant field we use that the integral of the x-component of the electrostatic force along the x-direction is exactly the electrostatic potential. If the integration is taken along the line $(x, y, z) = (-0.5 + t, 0, 0.5)$ with $0 \leq t \leq 1$ then the integral should be identically zero since the potential is periodic across the computational cube. A constant field can be added to enforce this result. After computing the x-component of the electrostatic force at each of the two opposite charges using the methodology outlined above (with 2 Chebyshev nodes in each direction, the values were compared to those obtained by Ewald summation. The numbers are summarized in Table I. The relative error in the bbFMM calculation is on the order of 10^{-6}.

7.4 Summary

In this chapter we described how the black-box fast multipole method presented in the preceding chapter can be extended to periodic systems. For absolutely convergent periodic sums this can be achieved by expanding the FMM tree to encompass the periodic images of the computational cells. Instead of organizing the images in an octree structure, they are gathered 27 cells at a time to allow for spatial symmetry. If the periodic sum is conditionally convergent, this extension of bbFMM is insufficient because the M2M operation leads to a divergent method. However we can remedy this difficulty by modifying the kernel so that the sum is absolutely convergent. In order to recover the original periodic sum, correction fields of different orders (e.g. constant, linear) need to be added. Finally we illustrated this procedure for a simple two-charge system and the results were compared to those obtained by Ewald summation.

Chapter 8

Torsion and Bending of Silicon Nanowires

We present a unified approach for atomistic modeling of torsion and bending of nanowires that is free from artificial end effects. Torsional and bending periodic boundary conditions (t-PBC and b-PBC) are formulated by generalizing the conventional periodic boundary conditions (PBC) to cylindrical coordinates. The approach is simpler than the more general Objective Molecular Dynamics formulation because we focus on the special cases of torsion and bending. A simple implementation of these boundary conditions is presented and correctly conserves linear and angular momenta. We also derive the Virial expressions for the average torque and bending moment under these boundary conditions that are analogous to the Virial expression for the average stress in PBC. The method is demonstrated by Molecular Dynamics simulation of Si nanowires under torsion and bending, which exhibit several modes of failure depending on their diameters.

8.1 Introduction

Recently there has been considerable interest in the directed growth of semiconductor nanowires (NWs), which can be used to construct nano-scale field effect transistors

(FETs) [17, 91, 45], chemical and biological sensors [16], nano-actuators [12] and nanofluidic components [31]. Epitaxially grown NWs have the potential to function as conducting elements between different layers of three-dimensional integrated circuits. Because significant stress may build up during fabrication and service (e.g. due to thermal or lattice mismatch), characterization and prediction of mechanical strength and stability of NWs is important for the reliability of these novel devices.

NWs also offer unique opportunities for studying the fundamental deformation mechanisms of materials at the nanoscale. The growing ability to fabricate and mechanically test microscale and nanoscale specimens and the increasing computational power allows for direct comparison between experiments and theory at the same length scale.

The size of these devices presents a challenge to test their mechanical properties. In macroscale samples, the materials are routinely tested in tension, shear, torsion and bending using standard grips and supports. Smaller samples, however, require more inventive testing techniques. For nanoscale testing, tensile and bending tests have been performed using nanoindentors, AFM [55, 24], and MEMS devices [95, 49]. Similar experiments have been performed at the microscale [52, 86]. With the rapid progress of nanofabrication and nanomanipulation capabilities, additional tension, torsion, and bending experimental data on crystalline and amorphous nanowires will soon be available.

Molecular dynamics is poised to be the main theoretical tool to help understand and predict small scale mechanical properties. However, since MD is limited in the number of atoms it can simulate; it cannot simulate whole nanowires. Either the nanowire simulated must be extremely short or periodic boundary conditions (PBC) must be used. End conditions artificially alter the material locally such that defect nucleation and failure often occurs there. This results in simulations that test the strength of the boundary rather than the intrinsic strength of the material. Traditional PBC remove this artifact by enforcing translational invariance and eliminating all artificial boundaries.

The use of conventional PBC allows for the simulation of tensile, pure shear, and simple shear in MD [75]. In fact, the mechanical properties of silicon nanowires in tension were recently calculated using this approach [54]. The nanowires were strained by extending the periodicity along the nanowire length and the stress was calculated through the Virial formula. However, regardless of the types of strain imposed on the periodic simulation cell, the images form a perfect lattice which precludes nonzero average torsion or bending. Therefore, to simulate torsion or bending tests, either small finite nanowires must be simulated or the current PBC framework must be altered.

Many Molecular Dynamics simulations on torsion and bending of nanoscale structures have been reported [44, 94, 74, 48, 71]. The artificial end effects are sometimes reduced by putting the ends far away from the region undergoing severe deformation, requiring a long nanowire [62]. There have also been attempts to rectify this problem [72]. Recently, the objective molecular dynamics (OMD) formulation [25] has been proposed that generalizes periodic boundary conditions to accommodate symmetries other than translational. Under this framework, torsion and bending simulations can be performed without end effects. But the general formulation of OMD is somewhat difficult to apply to existing MD simulation programs.

In this chapter, we present a simpler formulation that accommodates torsion and bending in a generalized periodic boundary condition framework. From this simple formulation we found that torsion and bending can be related to shear and normal strains when expressed in *cylindrical coordinates*. This leads to t-PBC and b-PBC, respectively, as formulated in Section 2. While only linear momenta are preserved in PBC, both t-PBC and b-PBC preserve the angular momentum around their rotation axes. These new boundary conditions can be easily implemented on top of existing simulation programs that use conventional PBC. In Section 3, we derive the Virial expressions for the torque and bending moment that are analogous to the Virial expressions for the average stress in simulation cells under PBC. The Virial expressions of torque and bending moment, expressed as a sum over discrete atoms, are found to correspond to a set of tensorial quantities in continuum mechanics, expressed as a volume integral. Section 4 presents the application of these new boundary conditions to modeling of the intrinsic strength of Si nanowires under torsion and bending.

8.2 Generalization of Periodic Boundary Conditions

8.2.1 Review of Conventional PBC

PBC can be visualized as a primary cell surrounded by a set of replicas, or image cells. The replicas are arranged into a regular lattice specified by three repeat vectors: $\mathbf{c}_1$, $\mathbf{c}_2$, $\mathbf{c}_3$. This means that whenever there is an atom at location $\mathbf{r}_i$ there are also atoms at $\mathbf{r}_i + n_1\mathbf{c}_1 + n_2\mathbf{c}_2 + n_3\mathbf{c}_3$, where n_1, n_2, n_3 are arbitrary integers [1, 11]. Because the atoms in the image cells behave identically as those in the primary cell, it is immaterial to specify which space belongs to the primary cell and which space belongs to the image cell. Even though it is customary to refer to the parallelepiped formed by the three period vectors as the simulation cell and the surface of this parallelepiped as the boundary, there is no physical interface at this boundary. In other words, the "boundary" between the primary and image cells in PBC can be drawn anywhere and is only a matter of convention. Consequently, translational invariance is preserved and linear momenta is conserved in all three directions. It is customary to set the velocity of the center of mass to zero in the initial condition which should remain zero during the simulation. This provides an important check of the self-consistency of the simulation program.

The scaled coordinates $\mathbf{s}_i$ are usually introduced to simplify the notation and the implementation of PBC, where

$$\mathbf{r}_i = \mathbf{H} \cdot \mathbf{s}_i \tag{8.1}$$

and $\mathbf{H} = [\mathbf{c}_1|\mathbf{c}_2|\mathbf{c}_3]$ is a 3×3 matrix whose three columns are formed by the coordinates of the three repeat vectors. For example, $\mathbf{H}$ becomes a diagonal matrix when the three repeat vectors are parallel to the x-, y-, z-axes, respectively,

$$\mathbf{H} = \begin{bmatrix} L_x & 0 & 0 \\ 0 & L_y & 0 \\ 0 & 0 & L_z \end{bmatrix} \tag{8.2}$$

where $L_x = |\mathbf{c}_1|$, $L_y = |\mathbf{c}_2|$, $L_z = |\mathbf{c}_3|$. The periodic boundary conditions can also be stated in terms of the scaled coordinates as follows: whenever there is an atom at location $\mathbf{s}_i = (s_x^i, s_y^i, s_z^i)^{\mathrm{T}}$, there are also atoms at location $(s_x^i + n_1, s_y^i + n_2, s_z^i + n_3)^{\mathrm{T}}$, where n_1, n_2, n_3 are arbitrary integers. The scaled coordinates of each atom, s_x^i, s_y^i, s_z^i are sometimes limited to $[-0.5, 0.5)$, although this is not necessary.

To apply a normal strain in the x direction, we only need to modify the magnitude of L_x. To introduce a shear strain ε_{yz}, we can simply add an off-diagonal term to the $\mathbf{H}$ matrix,

$$\mathbf{H} = \begin{bmatrix} L_x & 0 & 0 \\ 0 & L_y & 2\,\varepsilon_{yz}\,L_y \\ 0 & 0 & L_z \end{bmatrix} \tag{8.3}$$

Regardless of the normal or shear strain, the scaled coordinates, s_x^i, s_y^i, s_z^i, still independently satisfy PBC in the domain $[-0.5, 0.5)$, which is the main advantage for introducing the scaled coordinates. By modifying $\mathbf{H}$ in these ways, we can stretch and shear a crystal in MD.

8.2.2 Torsional PBC

While the exact formulation of PBC as stated above cannot accommodate a non-zero average torsion over the entire simulation cell, the general idea can still be used. Consider a nanowire of length L_z aligned along the z-axis, as shown in Fig. 8.1(a). To apply PBC along the z-axis, we can make two copies of the atoms in the nanowire, shift them along z by $\pm L_z$, and let them interact with the atoms in the primary wire. Two copies of the original nanowire would be sufficient if the cut-off radius r_c of the interatomic potential function is smaller than L_z (usually $r_c \ll L_z$). After PBC is applied, the model may be considered as an infinitely long, periodic wire along the z-axis. Any arbitrary section of length L_z can now be considered as the primary wire due to the periodicity. Since the atomic arrangement must repeat itself after every L_z distance along the wire, the average torsion we can apply to the nanowire is zero. A local torsion in some section of the wire has to be cancelled by an opposite torsion at another section that is less than L_z away.

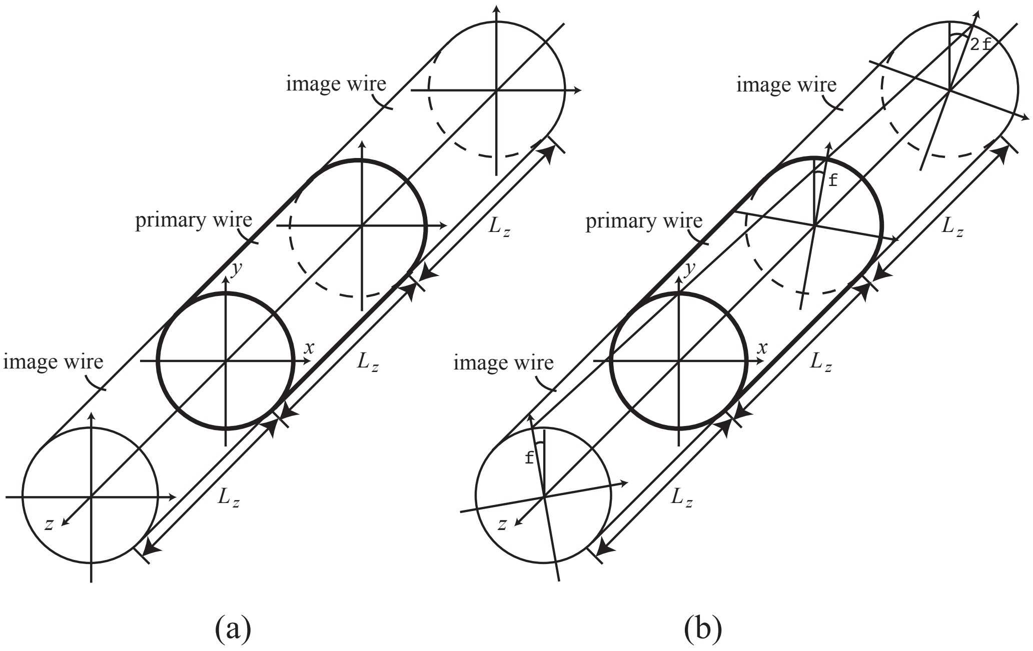


Figure 8.1: (a) A nanowire subjected to PBC along z axis. (b) A nanowire subjected to t-PBC along z axis.

One way to introduce an average torque to this infinitely long wire is to rotate the two images by angle $+\phi$ and $-\phi$, respectively, before we attach them to the two ends of the primary wire as shown in Fig. 8.1(b). The image wire that is displaced by L_z is rotated by ϕ, while the one that is displaced by $-L_z$ is rotated by $-\phi$. In this case, as we travel along the wire by L_z, we will find that the atomic arrangement in the cross section will be rotated around z axis by angle ϕ but otherwise identical. Again, because this property is satisfied by any cross section of the nanowire, it is arbitrary which we call the primary wire and which we call images similar to conventional periodic boundary conditions. The torsion imposed on the nanowire can be characterized by the angle of rotation per unit length, ϕ/L_z. In the limit of small deformation, the shear strain field produced by the torsion is,

$$\varepsilon_{\theta z} = \frac{r\,\phi}{2\,L_z} \tag{8.4}$$

where r is the distance away from the z-axis.

The above procedure specifies torsional periodic boundary conditions (t-PBC) that can be easily expressed in terms of scaled cylindrical coordinates. Consider an atom i with cartesian coordinates $\mathbf{r}_i = (x_i, y_i, z_i)^{\mathrm{T}}$ and cylindrical coordinates $(r_i, \theta_i, z_i)^{\mathrm{T}}$, which are related to each other by,

$$\begin{aligned} x_i &= r_i \cos\theta_i && (8.5) \\ y_i &= r_i \sin\theta_i && (8.6) \end{aligned}$$

When the wire is subjected to PBC along z (with free boundary conditions in x and y), we introduce the scaled cylindrical coordinates $(s_r^i, s_\theta^i, s_z^i)^{\mathrm{T}}$ through the relationship

$$\begin{pmatrix} r_i \\ \theta_i \\ z_i \end{pmatrix} = \begin{bmatrix} R & 0 & 0 \\ 0 & 2\pi & 0 \\ 0 & 0 & L_z \end{bmatrix} \begin{pmatrix} s_r^i \\ s_\theta^i \\ s_z^i \end{pmatrix} \equiv \mathbf{M} \cdot \begin{pmatrix} s_r^i \\ s_\theta^i \\ s_z^i \end{pmatrix} \tag{8.7}$$

Both s_θ^i and s_z^i independently satisfy periodic boundary conditions in the domain $[-0.5, 0.5)$. No boundary condition is applied to coordinate s_r^i. R is a characteristic length scale in the radial direction in order to make s_r^i dimensionless. Although this is not necessary, one can choose R to be the radius of the nanowire, in which case s_r^i would vary from 0 to 1.

Torsion can be easily imposed by introducing an off-diagonal term to the matrix $\mathbf{M}$, which becomes

$$\mathbf{M} = \begin{bmatrix} R & 0 & 0 \\ 0 & 2\pi & \phi \\ 0 & 0 & L_z \end{bmatrix} \tag{8.8}$$

The scaled coordinates, s_θ^i and s_z^i, still independently satisfy periodic boundary conditions in the domain $[-0.5, 0.5)$. This is analogous to the application of shear strain to a simulation cell subjected to conventional PBC, as described in Eq. (8.3). t-PBC can be easily implemented in an existing simulation program by literally following Fig. 8.1(b), i.e. by making two copies of the wire, rotating them by $\pm\phi$, and placing the two copies at the two ends of the primary wire. In practice, it is not necessary to copy the entire wire, because the cut-off radius r_c of the interatomic potential function is usually much smaller than L_z. Only two sections at the ends of the primary wire with lengths longer than r_c need to be copied.[1] It is important to perform this operation of "copy-and-paste" at every MD time step, or whenever the potential energy and atomic forces need to be evaluated. This will completely remove the end effects and will ensure that identical MD trajectories will be generated had we chosen a different section (also of length L_z) of the wire as our primary wire.

An important property of the t-PBC is that the trajectory of every atom satisfy the classical (Newton's) equation of motion. In other words, among the infinite number of atoms that are periodic images of each other, it makes no physical difference as to which one should be called "primary" and which ones should be called "images". Since the primary atoms follow the Newton's equation of motion ($\mathbf{f}_i = m\,\mathbf{a}_i$), to prove the above claim it suffices to show that the image atoms, which are slaves of the primary atoms (through the "copy-and-paste" operation) also follow the Newton's

[1] This simple approach is not able to accommodate long-range Coulomb interactions, for which the Ewald summation is usually used in conventional PBC. Extension of the Ewald method to t-PBC is beyond the scope of this paper.

equation of motion ($\mathbf{f}_{i'} = m\,\mathbf{a}_{i'}$).

To show this, consider an atom i and its periodic image i', such that $s_r^{i'} = s_r^i$, $s_\theta^{i'} = s_\theta^i$, $s_z^{i'} = s_z^i + 1$. The position of the two atoms are related by t-PBC: $\mathbf{r}_{i'} = \mathrm{Rot}_z(\mathbf{r}_i, \phi) + \hat{\mathbf{e}}_z\, L_z$, where $\mathrm{Rot}_z(\cdot, \phi)$ represent rotation of a vector around z-axis by angle ϕ and $\hat{\mathbf{e}}_z$ is the unit vector along z-axis. Hence the acceleration of the two atoms are related to each other through: $\mathbf{a}_{i'} = \mathrm{Rot}_z(\mathbf{a}_i, \phi)$. Now consider an arbitrary atom j that falls within the cut-off radius of atom i. Let $\mathbf{r}_{ij} \equiv \mathbf{r}_j - \mathbf{r}_i$ be the distance vector from atom i to j. Consider the image atom j' such that $s_r^{j'} = s_r^j$, $s_\theta^{j'} = s_\theta^j$, $s_z^{j'} = s_z^j + 1$. Hence $\mathbf{r}_{j'} = \mathrm{Rot}_z(\mathbf{r}_j, \phi) + \hat{\mathbf{e}}_z\, L_z$, and $\mathbf{r}_{i'j'} \equiv \mathbf{r}_{j'} - \mathbf{r}_{i'} = \mathrm{Rot}_z(\mathbf{r}_{ij}, \phi)$. Since this is true for an arbitrary neighbor atom j around atom i, the forces on atoms i and i' must satisfy the relation: $\mathbf{f}_{i'} = \mathrm{Rot}_z(\mathbf{f}_i, \phi)$. Therefore, the trajectory of atom i' also satisfies the Newton's equation of motion $\mathbf{f}_{i'} = m\,\mathbf{a}_{i'}$.

MD simulations under t-PBC should conserve the total linear momentum P_z and angular momentum J_z because t-PBC preserves both translational invariance along and rotational invariance around the z axis. However, the linear momenta P_x and P_y are no longer conserved in t-PBC due to the specific choice of the origin in the x-y plane (which defines the cylindrical coordinates r and θ). In comparison, the angular momentum J_z is usually not conserved in PBC. Consequently, at the beginning of MD simulations under t-PBC, we need to set both P_z and J_z to zero. P_z and J_z will remain zero, which provides an important self-consistency check of the implementation of boundary conditions and numerical integrators.

8.2.3 Bending PBC

The same idea can be used to impose bending deformation on wires. Again, we will describe the atomic positions through scaled cylindrical coordinates, $(s_r^i, s_\theta^i, s_z^i)^{\mathrm{T}}$, which is related to the real cylindrical coordinates, $(r^i, \theta^i, z^i)^{\mathrm{T}}$, through the following transformation,

$$\begin{pmatrix} r_i \\ \theta_i \\ z_i \end{pmatrix} = \begin{bmatrix} R & 0 & 0 \\ 0 & \Theta & 0 \\ 0 & 0 & L_z \end{bmatrix} \begin{pmatrix} s_r^i \\ s_\theta^i \\ s_z^i \end{pmatrix} + \begin{pmatrix} L_0/\Theta \\ 0 \\ 0 \end{pmatrix} \equiv \mathbf{N} \cdot \begin{pmatrix} s_r^i \\ s_\theta^i \\ s_z^i \end{pmatrix} + \begin{pmatrix} L_0/\Theta \\ 0 \\ 0 \end{pmatrix} \tag{8.9}$$

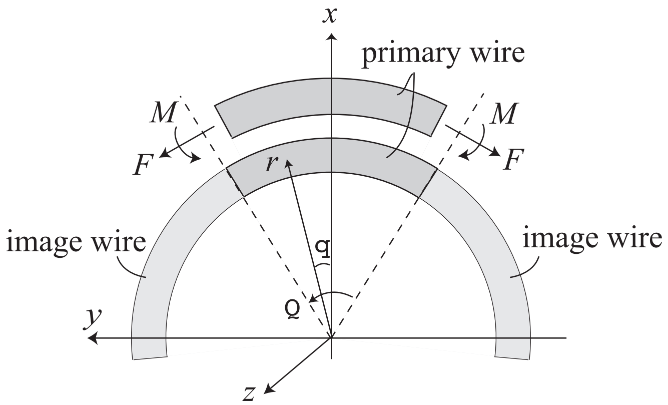


Figure 8.2: A nanowire subjected to b-PBC around z axis. At equilibrium the net line tension force F must vanish but a non-zero bending moment M will remain.

While the coordinate system here is still the same as that in the case of torsion, the wire is oriented along the θ direction, as shown in Fig. 8.2. Among the three scaled coordinates, only s_θ^i is subjected to a periodic boundary condition, in the domain of $[-0.5, 0.5)$. This means that θ^i is periodic in the domain $[-\Theta/2, \Theta/2)$. No boundary conditions are applied to s_r^i and s_z^i. R and L_z are characteristic length scales in the r and z directions, respectively. L_0 is the original (stress free) length of the wire and $\rho = L_0/\Theta$ is the radius of curvature of the wire. The equation $r = \rho$ specifies the neutral surface of the wire. Thus, $r_i = \rho + R\, s_r^i$, where $R\, s_r^i$ describes the displacement of atom i away from the neutral axis in the r direction.

In the previous section, an off-diagonal element has to be introduced to the transformation matrix $\mathbf{M}$ in order to introduce torsion. In comparison, the form of Eq. (8.9) does not need to be changed to accommodate bending. Different amount of bending can be imposed by adjusting the value Θ, while the matrix $\mathbf{N}$ remains diagonal. The larger Θ is the more severe the bending deformation. The state of zero bending corresponds to the limit of $\Theta \to 0$.

Intuitively, it may seem that increasing the value of Θ would elongate the wire and hence induce a net tension force F in addition to a bending moment M. However, this is not the case because the direction of force F at the two ends of the wire are not parallel to each other, as shown in Fig. 8.2. When no lateral force (i.e. in the r direction) is applied to the wire, F must vanish for the entire wire to reach equilibrium. Otherwise, there will be a non-zero net force in the $-x$ direction, which will cause the wire to move until F become zero. At equilibrium, only a bending moment (but no tension force) can be imposed by b-PBC.

b-PBC can be implemented in a similar way as t-PBC. We make two copies of the primary wire and rotate them around the z axis by $\pm\Theta$. The atoms in these copies will interact and provide boundary conditions for atoms in the primary wire.[2] Again, this "copy-and-paste" operation is required at every step of MD simulation. This will ensure all atoms (primary and images) satisfy Newton's equation of motion. The proof is similar to that given in the previous section for t-PBC and is omitted here for brevity. Interestingly, both the linear momentum P_z and the angular momentum J_z for the center of mass are conserved in b-PBC, exactly the same as t-PBC. Therefore, both P_z and J_z must be set to zero in the initial condition of MD simulations.

8.3 Virial Expressions for Torque and Bending Moment

The experimental data on tensile tests are usually presented in the form of stress-strain curves. The normal stress is calculated from, $\sigma = F/A$, where F is the force applied to the ends and A is the cross section area of the wire. In experiments on macroscopic samples, the end effects are reduced by making the ends of the specimen much thicker than the middle (gauge) section where significant deformation is expected. In atomistic simulations, on the other hand, the end effects are removed by a different approach, usually through the use of periodic boundary conditions. Unfortunately, with the end effects completely removed by PBC, there is no place to serve as grips where external forces can be applied. Therefore, the stress must be

[2] Similar to the case of t-PBC, this simple approach is not able to accommodate long-range Coulomb interactions. While a wire under t-PBC can be visualized as an infinitely long wire, this interpretation will encounter some difficulty in b-PBC, because continuing the curved wire along the θ-direction will eventually make the wire overlap. The interpretation of b-PBC would then require the wire to exist in a multi-sheeted Riemann space [84, page 80] so that the wire does not really overlap with each other.

computed differently in atomistic simulations under PBC than in experiments. The Virial stress expression is widely used in atomistic calculations, which represents the time and volume average of stress in the simulation cell.

The same problem appears in atomistic simulations under t-PBC and b-PBC. There needs to be a procedure to compute the torque and bending moment in these new boundary conditions. In this section, we develop Virial expressions for the torque and bending moment in t-PBC and b-PBC. Similar to the Virial stress, the new expressions involve discrete sums over all atoms in the simulation cell. The corresponding expressions in continuum mechanics, expressed in terms of volume integrals, are also identified. Since the derivation of these new expressions are motivated by that of the original Virial expression, we will start with a quick review of the Virial stress.

8.3.1 Virial Stress in PBC

For an atomistic simulation cell subjected to PBC in all three directions, the Virial formula gives the stress averaged over the entire simulation cell at thermal equilibrium as

$$\sigma_{\alpha\beta} = \frac{1}{\Omega}\left\langle \sum_{i=1}^{N} -m_i\, v_\alpha^i\, v_\beta^i + \sum_{i=1}^{N-1}\sum_{j=i+1}^{N} \frac{\partial V}{\partial(x_\alpha^i - x_\alpha^j)}(x_\beta^i - x_\beta^j) \right\rangle \tag{8.10}$$

In this formula $\Omega = \det(\mathbf{H})$ is the volume of the simulation cell, N is the total number of atoms, v_α^i and x_α^i are the α-components of the velocity and position of atom i, and V is the potential energy. The terms $(x_\alpha^i - x_\alpha^j)$ and $(x_\beta^i - x_\beta^j)$ in the second summation are assumed to be taken from the nearest images of atom i and atom j. The bracket $\langle\cdot\rangle$ means ensemble average, which equals to the long time average if the system has reached equilibrium. Thus the Virial stress is the stress both averaged over the entire space and over a long time.

The Virial stress is the derivative of the free energy F of the atomistic simulation cell with respect to a virtual strain $\varepsilon_{\alpha\beta}$, which deforms the periodic vectors $\mathbf{c}_1$, $\mathbf{c}_2$ and $\mathbf{c}_3$ and hence the matrix $\mathbf{H}$,

$$\sigma_{\alpha\beta} \;=\; \frac{1}{\Omega}\frac{\partial F}{\partial \varepsilon_{\alpha\beta}} \tag{8.11}$$

Assuming the simulation cell is in equilibrium under the canonical ensemble, the free energy is defined as,

$$F \equiv -k_B T \ln \left\{ \frac{1}{h^{3N} N!} \int d^{3N}\mathbf{r}_i d^{3N}\mathbf{p}_i \exp\left[-\frac{1}{k_B T} \left(\sum_{i=1}^{N} \frac{|\mathbf{p}_i|^2}{2m_i} + V(\{\mathbf{r}_i\}) \right) \right] \right\} \tag{8.12}$$

where k_B is the Boltzmann's constant, T is temperature, h is Planck's constant, $\mathbf{r}_i$ and $\mathbf{p}_i$ are atomic position and momentum vectors, and V is the interatomic potential function. The momenta can be integrated out explicitly to give,

$$F = -k_B T \ln \left\{ \frac{1}{\Lambda^{3N} N!} \int d^{3N}\mathbf{r}_i \exp\left[-\frac{V(\{\mathbf{r}_i\})}{k_B T} \right] \right\} \tag{8.13}$$

where $\Lambda \equiv h/(2\pi m k_B T)^{1/2}$ is the thermal de Broglie wavelength. In atomistic simulations under PBC, the potential energy can be written as a function of the scaled coordinates $\{\mathbf{s}_i\}$ and matrix H. Hence, F can also be written in terms of an integral over the scaled coordinates.

$$F = -k_B T \ln \left\{ \frac{\Omega^N}{\Lambda^{3N} N!} \int d^{3N}\mathbf{s}_i \exp\left[-\frac{V(\{\mathbf{s}_i\}, \mathbf{H})}{k_B T} \right] \right\} \tag{8.14}$$

The Virial formula can be obtained by taking the derivative of Eq. (8.14) with respect to $\epsilon_{\alpha\beta}$. The first term in the Virial formula comes from the derivative of the volume Ω with respect to $\varepsilon_{\alpha\beta}$, which contributes a $-N\, k_B\, T\, \delta_{\alpha\beta}/\Omega$ term to the total stress. This is equivalent to the velocity term in the Virial formula because $\left\langle m_i\, v^i_\alpha\, v^i_\beta \right\rangle = k_B\, T\, \delta_{\alpha\beta}$ in the canonical ensemble. The second term comes from the derivative of the potential energy $V(\{\mathbf{s}_i\}, \mathbf{H})$ with respect to $\varepsilon_{\alpha\beta}$. The Virial stress expression can also be derived in several alternative approaches (see [89, 64, 14, 15, 96] for more discussions). The corresponding quantity for Virial stress in continuum mechanics is the volume average of the stress tensor,

$$\overline{\sigma}_{ij} = \frac{1}{\Omega} \int_\Omega \sigma_{ij}\, dV = \frac{1}{\Omega} \oint_S t_j\, x_i\, dS \tag{8.15}$$

where the integral $\oint_S$ is over the bounding surface of volume Ω, t_j is the traction force density on surface element dS, and x_i is the position vector of the surface element.

8.3.2 Virial Torque in t-PBC

The Virial torque expression for a simulation cell subjected to t-PBC can be derived in a similar fashion. First, we re-write the potential energy V as a function of the scaled cylindrical coordinates and the components of matrix $\mathbf{M}$, as given in Eq. (8.8),

$$V(\{\mathbf{r}_i\}) = V(\{s_r^i, s_\theta^i, s_z^i\}, R, \phi, L_z) \tag{8.16}$$

The Virial torque is then defined as the derivative of the free energy F with respect to ϕ,

$$\begin{aligned} \tau &\equiv \frac{\partial F}{\partial \phi} && (8.17) \\ F &= -k_B T \ln \left\{ \frac{\Omega^N}{\Lambda^{3N} N!} \int d^{3N}\mathbf{s}_i \exp\left[-\frac{V(\{s_r^i, s_\theta^i, s_z^i\}, R, \phi, L_z)}{k_B T} \right] \right\} && (8.18) \end{aligned}$$

Since $\partial\Omega/\partial\phi = 0$, the torque reduces to

$$\tau = \frac{\int d^{3N}\mathbf{s}_i \exp\left[-\frac{V(\{s_r^i, s_\theta^i, s_z^i\}, R, \phi, L_z)}{k_B T} \right] \frac{\partial V}{\partial \phi}}{\int d^{3N}\mathbf{s}_i \exp\left[-\frac{V(\{s_r^i, s_\theta^i, s_z^i\}, R, \phi, L_z)}{k_B T} \right]} \equiv \left\langle \frac{\partial V}{\partial \phi} \right\rangle \tag{8.19}$$

In other words, the torque τ is simply the ensemble average of the derivative of the potential energy with respect to torsion angle ϕ. To facilitate calculation in an atomistic simulation, we can express $\frac{\partial V}{\partial \phi}$ in terms of the real coordinates of the atoms,

$$\frac{\partial V}{\partial \phi} = \frac{1}{L_z} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} -\frac{\partial V}{\partial (x_i - x_j)} (y_i\, z_i - y_j\, z_j) + \frac{\partial V}{\partial (y_i - y_j)} (x_i\, z_i - x_j\, z_j) \tag{8.20}$$

Hence we arrive at the Virial torque expression

$$\tau = \frac{1}{L_z} \left\langle \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} -\frac{\partial V}{\partial (x_i - x_j)} (y_i\, z_i - y_j\, z_j) + \frac{\partial V}{\partial (y_i - y_j)} (x_i\, z_i - x_j\, z_j) \right\rangle \tag{8.21}$$

There is no velocity term in Eq. (8.21) because modifying ϕ does not change the volume Ω of the wire. This expression is verified numerically in Appendix C in the zero temperature limit when the free energy equals to the potential energy. The corresponding quantity in continuum elasticity theory can be written in terms of an integral over the volume Ω of the simulation cell,

$$\tau = Q_{zz} \equiv \frac{1}{L_z} \int_{\Omega} -y\, \sigma_{xz} + x\, \sigma_{yz}\, dV \tag{8.22}$$

The derivation is given in Appendix A. The stress in the above expression refers to the Cauchy stress in the context of finite deformation. Because it uses current coordinates, the expression remains valid in finite deformation. The correspondence between Eqs. (8.21) and (8.22) bears a strong resemblance to the correspondence between Eqs. (8.10) and (8.15). While the Virial stress formula corresponds to the average (i.e. zeroth moment) of the stress field over volume Ω, τ corresponds to a linear combination of the first moments of the stress field.

8.3.3 Virial Bending Moment in b-PBC

Following a similar procedure, we can obtain the Virial expression for the bending moment for a simulation cell subjected to b-PBC. First, we rewrite the potential energy of a system under b-PBC as,

$$V(\{\mathbf{r}_i\}) = V(\{s_r^i, s_\theta^i, s_z^i\}, R, \Theta, L_z) \tag{8.23}$$

The Virial bending moment is then the derivative of the free energy with respect to Θ.

$$\begin{aligned} M &\equiv \frac{\partial F}{\partial \Theta} && (8.24)\\ F &= -k_B T \ln \left\{ \frac{\Omega^N}{\Lambda^{3N} N!} \int d^{3N}\mathbf{s}_i \exp\left[-\frac{V(\{s_r^i, s_\theta^i, s_z^i\}, R, \Theta, L_z)}{k_B T} \right] \right\} && (8.25) \end{aligned}$$

Again, we find that M is simply the ensemble average of the derivative of potential energy with respect to Θ,

$$M = \left\langle \frac{\partial V}{\partial \Theta} \right\rangle \tag{8.26}$$

The derivative $\frac{\partial V}{\partial \Theta}$ can be expressed in terms of the real coordinates of the atoms,

$$\begin{aligned}\frac{\partial V}{\partial \Theta} &= \tfrac{1}{\Theta}\textstyle\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} \quad -\frac{\partial V}{\partial (x_i - x_j)}(y_i\,\theta_i - y_j\,\theta_j + \rho\,\cos\theta_i - \rho\,\cos\theta_j) \\ &\qquad + \frac{\partial V}{\partial (y_i - y_j)}(x_i\,\theta_i - x_j\,\theta_j - \rho\,\sin\theta_i + \rho\,\sin\theta_j) \quad (8.27)\end{aligned}$$

Hence we arrive at the Virial bending moment expression,

$$\begin{aligned}M &= \tfrac{1}{\Theta}\Big\langle\textstyle\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} \quad -\frac{\partial V}{\partial (x_i - x_j)}\,(y_i\,\theta_i - y_j\,\theta_j + \rho\,\cos\theta_i - \rho\,\cos\theta_j) \\ &\qquad + \frac{\partial V}{\partial (y_i - y_j)}\,(x_i\,\theta_i - x_j\,\theta_j - \rho\,\sin\theta_i + \rho\,\sin\theta_j)\Big\rangle (8.28)\end{aligned}$$

There is no velocity term in Eq. (8.28) because modifying Θ does not change the volume Ω of the wire. This expression is verified numerically in Appendix D in the zero temperature limit when the free energy equals to the potential energy. The corresponding quantity in continuum elasticity theory can be written in terms of an integral over the volume Ω of the simulation cell,

$$\begin{aligned}M &= Q_{z\theta} = \frac{1}{\Theta}\int_A dA \int_0^{\Theta} d\theta\,(-y\,\sigma_{x\theta} + x\,\sigma_{y\theta}) \qquad (8.29)\\ &= \frac{1}{\Theta}\int_A dA \int_0^{\Theta} d\theta\; r\,\sigma_{\theta\theta} = \frac{1}{\Theta}\int_{\Omega} \sigma_{\theta\theta}\,dV\end{aligned}$$

where A is the cross-section area of the continuum body subjected to b-PBC. The correspondence between Eqs. (8.28) and (8.30) bears a strong resemblance to the correspondence between Eqs. (8.10) and (8.15). Similar to τ, M also corresponds to a linear combination of the first moments of the stress field over the simulation cell volume.

8.4 Numerical Results

In this section, we demonstrate the usefulness of t-PBC and b-PBC described above by torsion and bending Molecular Dynamics simulations of Si nanowires (NWs) to failure.

Figure 8.3: Snapshots of Si NWs of two diameters before torsional deformation and after failure. The failure mechanism depends on its diameter.

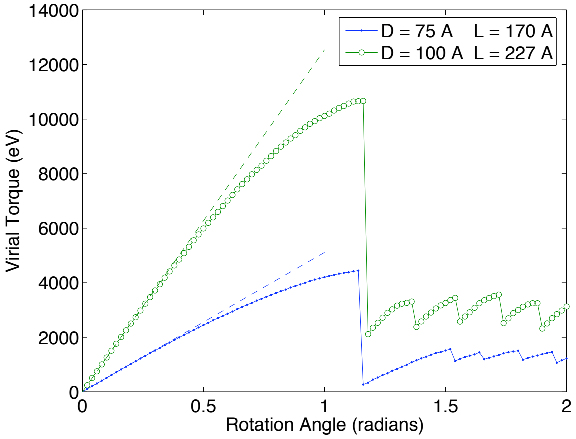


Figure 8.4: Virial torque τ as a function of rotation angle ϕ between the two ends of the NWs of two different diameters. Because the two NWs have the same aspect ratio L_z/D, they have the same maximum strain (on the surface) $\gamma_{\max} = \frac{\phi D}{2L_z}$ at the same twist angle ϕ.

The interactions between Si atoms are described by the modified-embedded-atom-method (MEAM) potential [5], which has been found to be more reliable in the study of the failure Si NWs than several other potential models for Si [54]. We considered two NWs both oriented along the [111] direction with diameters $D = 7.5$ nm and $D = 10$ nm and the same aspect ratio $L_z/D = 2.27$. To make sure the NW surface is well reconstructed, the NWs are annealed by MD simulations at 1000 K for 1 ps followed by a conjugate gradient relaxation. To save space, we only present simulation results on initially perfect NWs under torsion and bending deformation at $T = 300$ K. The effects of temperature and initial surface defects on the failure behavior of Si NWs will be presented in a subsequent paper.

8.4.1 Si Nanowire under Torsion

Simulations of Si NWs under torsion can be carried out easily using t-PBC. Before applying a torsion, we first equilibrate the NWs at the specified temperature and zero stress (i.e. zero axial force) by MD simulations under PBC where the NW length is allowed to elongate to accommodate the thermal strain. Fig. 8.3(a) and (c) shows the annealed Si NW structures. Subsequently, torsion is applied to the NW through t-PBC, where the twist angle ϕ (between two ends of the NW) increases in steps of 0.02 radian ($\approx 1.15^\circ$). For each twist angle, MD simulation under t-PBC is performed for 2 ps. The Nose-Hoover thermostat is used to maintain the temperature at $T = 300$ K using the Stömer-Verlet time integrator [10] with a time step of 1 fs. The linear momentum P_z and angular momentum J_z are conserved within $2\times10^{-10}\mathrm{eV}\cdot\mathrm{ps}\cdot\mathring{A}^{-1}$ and $9\times10^{-7}\mathrm{eV}\cdot\mathrm{ps}$, during the simulation, respectively. The twist angle continues to increase until the NW fails. If the Virial torque at the end of the 2 ps simulation is lower than that at the beginning of the simulation, the MD simulation is continued in 2 ps increments without increasing the twist angle, until the bending moment increases. The purpose of this approach is to give enough simulation time to resolve the failure process whenever that occurs. The Virial torque is computed by time averaging over the last 1 ps of the simulation for each twist angle. The torque versus twist angle relationship is plotted in Fig. 8.4.

The τ-ϕ curve is linear for small values of ϕ and becomes non-linear as ϕ approaches the critical value at failure. The torsional stiffness can be obtained from the torque-twist relationship and its value at small ϕ can be compared to theory. The torsional stiffness is defined as

$$k_\mathrm{t} \equiv \frac{\partial \tau}{\partial \phi} \tag{8.30}$$

In the limit of $\phi \to 0$, the torsional stiffness is estimated to be $k_\mathrm{t} = 5.11\times10^3$ eV for $D = 7.5$ nm and $k_\mathrm{t} = 1.25\times10^4$ eV for $D = 10$ nm. Strength of Materials predicts the following relationships for elastically isotropic circular shafts under torsion:

$$\tau = \frac{\phi}{L_z} G\,J\ , \qquad k_\mathrm{t} = \frac{G\,J}{L_z} \tag{8.31}$$

where G is the shear modulus, $J = \pi D^4/32$ is the polar moment of inertia. We note that this expression is valid only in the limit of small deformation ($\phi \to 0$). To compare our simulation results against this expression, we need to use the shear modulus of Si given by the MEAM model ($C_{11} = 163.78$ GPa, $C_{12} = 64.53$ GPa,

$C_{44} = 76.47$ GPa) on the (111) plane, which is $G = 58.57$ GPa. The predictions of the torsional stiffness from Strength of Materials are compared with the estimated value from MD simulations in Table I. The predictions overestimate the MD results by $25 \sim 30\%$. However, this difference can be easily eliminated by a slight adjustment ($\sim 6\%$) of the NW diameter D, given that $k_{\rm t} \propto D^4$. The adjusted diameters D^* for the two NWs is approximately $6\,\mathring{A}$ smaller than the nominal diameters D, which corresponds to a reduction of the NW radius by $3\,\mathring{A}$. This can be easily accounted for by the inaccuracy in the definition of NW diameter and the possibility of a weak surface layer on Si NWs [54].

Table I: Comparison of torsional stiffness for Si NW estimated from MD simulations and that predicted by Strength of Materials (SOM) theory. D^* is the adjusted NW diameter that makes the SOM predictions exactly match MD results. The critical twist angle ϕ_c and critical shear strain γ_c at failure are also listed.

Nominal diameter D	$k_{\rm t}$ (MD)	$k_{\rm t}$ (SOM)	Adjusted diameter D^*	ϕ_c	γ_c
7.5 nm	5110 eV	6680 eV	7.0 nm	1.16 (rad)	0.26
10.0 nm	12538 eV	15812 eV	9.4 nm	1.18 (rad)	0.26

The above agreement gives us confidence in the use of Strength of Materials to describe the behavior of NWs under torsion. Hence, we will use it to extract the critical strain in both NWs at failure. The maximum strain (engineering strain) in a cylindrical torsional shaft occurs on its surface,

$$\gamma_{\rm max} = \frac{\phi\, D}{2\, L_z} \tag{8.32}$$

Given that the aspect ratio of NWs is kept at $L_z/D = 2.27$, we have

$$\gamma_{\rm max} = 0.22\,\phi \tag{8.33}$$

for both NWs. The critical twist angle and critical strain at failure for both NWs are listed in Table I.

We expect the critical shear strain at failure to be independent of the shaft diameter for large diameters. This seems to hold remarkably well in our NW torsion simulation. Because the NW under t-PBC has no “ends”, failure can initiate anywhere along the NW. However, different failure mechanism are observed in the two NWs with different diameters. The thinner NW fails by sliding along a (111) plane, as seen in Fig. 8.3(b). The thicker NW fails by sliding both along a (111) plane and along longitudinal planes, creating wedges on the (111) cross section, as seen in Fig. 8.3(d). The failure mechanism of the thicker NW is also more gradual than that of the thinner NW. As can be observed in Fig. 8.4, the torque is completely relieved on the thinner NW when failure occurs, whereas the thicker NW experiences a sequence of failures. A more detailed analysis on the size dependence of NW failure modes and their mechanisms will be presented in a subsequent paper.

8.4.2 Si Nanowire under Bending

Simulations of Si NWs can be carried out using b-PBC just as we did for torsion. The Si NWs are equilibrated in the same way as described in the previous section before we apply bending through b-PBC. The bending angle Θ (between two ends of the NW) increases in steps of 0.02 radian ($\approx 1.15^\circ$). For each twist angle, MD simulations under b-PBC were performed for 2 ps. The linear momentum P_z and angular momentum J_z is conserved to the same level of precision as in the torsion simulations. The bending angle continues to increase until the NW fails. If the Virial bending moment at the end of the 2 ps simulation is lower than that at the beginning of the simulation, the MD simulation is continued in 2 ps increments without increasing the bending angle, until the bending moment increases. The purpose of this approach is to give enough simulation time to resolve the failure process whenever that occurs. The Virial bending moment is computed by a time average over the last 1 ps of the simulation for each twist angle. The bending moment versus bending angle relationship is plotted in Fig. 8.5.

The M-Θ curve is linear for small values of Θ and becomes non-linear as Θ approaches the critical value at failure. The bending stiffness can be computed from

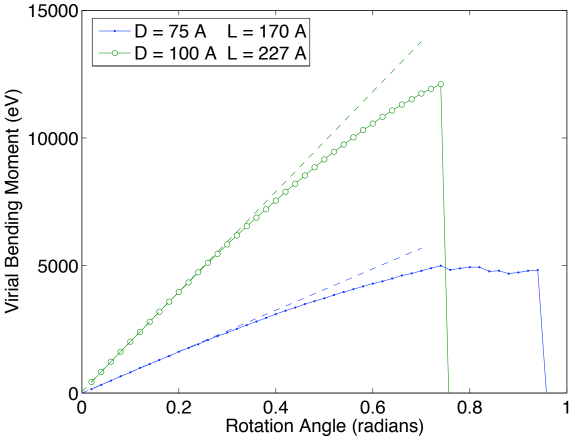


Figure 8.5: Virial bending moment M as a function of bending angle Θ between the two ends of the two NWs with different diameters. Because the two NWs have the same aspect ratio L_z/D, they have the same maximum strain $\epsilon_{\max} = \frac{\Theta D}{2L_z}$ at the same bending angle Θ.

the M-Θ curve and its value at small Θ can be compared to theory. Similar to the torsional stiffness in the previous section, we define a bending stiffness as

$$k_{\rm b} \equiv \frac{\partial M}{\partial \Theta} \tag{8.34}$$

In the limit of $\Theta \to 0$ the bending stiffness is estimated to be $k_{\rm b} = 8.12 \times 10^3$ eV for $D = 7.5$ nm and $k_{\rm b} = 1.96 \times 10^4$ eV for $D = 10$ nm. Strength of Materials predicts the following relationships for elastically isotropic beam under bending,

$$M = \frac{\Theta}{L_0} E\, I_z \ , \qquad k_{\rm b} = \frac{E\, I_z}{L_0} \tag{8.35}$$

where E is the Young's modulus, $I_z = \pi D^4/64$ is the moment of inertia of the NW cross section around z-axis. To compare our simulation results against this expression, we need to use the Young's modulus of Si given by the MEAM model along the [111] direction, which is 181.90 GPa. The predictions of the torsional stiffness from Strength of Materials are compared with the estimated value from MD simulations in Table II. The predictions overestimate the MD results by $23 \sim 25\%$. But this difference can be easily eliminated by a slight adjustment ($\sim 5\%$) of the NW diameter D, given that

$k_{\mathrm{b}} \propto D^4$. The adjusted diameters D^* for the two NWs is approximately $5\,\mathring{A}$ smaller than the nominal diameters D, which corresponds to a reduction of the NW radius by $2.5\,\mathring{A}$. It is encouraging to see that the adjusted diameters from torsion simulations match those for the bending simulations reasonably well.

Table II: Comparison of the bending stiffnesses for Si NWs estimated from MD simulations and that predicted by Strength of Materials (SOM) theory. D^* is the adjusted NW diameter that makes SOM predictions exactly match MD results. The critical bending angle Θ_f and critical normal strain ϵ_f at fracture are also listed.

Nominal diameter D	k_{b} (MD)	k_{b} (SOM)	Adjusted diameter D^*	Θ_f	ϵ_f
7.5 nm	8117 eV	10374 eV	7.1 nm	0.96 (rad)	0.21
10.0 nm	19619 eV	24554 eV	9.5 nm	0.76 (rad)	0.17

The above agreement gives us confidence in the use of Strength of Materials theory to describe the behavior of NW under bending. Hence we will use it to extract the critical strain experienced by both NWs at the point of fracture. Based on the Strength of Materials theory, the maximum strain (engineering strain) of a beam in pure bending occurs at the points furthest away from the bending axis,

$$\epsilon_{\mathrm{max}} = \frac{\Theta\, D}{2\, L_0} \tag{8.36}$$

Since the aspect ratio of NWs is kept at $L_0/D = 2.27$, we have

$$\epsilon_{\mathrm{max}} = 0.22\,\Theta \tag{8.37}$$

for both NWs. The critical bending angle and critical normal strain at failure for both NWs are listed in Table II. The critical strain at fracture is similar to results obtained from MD simulations of Si NWs under uniaxial tension, $\epsilon_f = 0.18$, also using the MEAM model [54]. The higher critical stress value observed in the thinner NW in bending is related to the higher stress gradient across its cross section.

Fig. 8.6 shows the atomic structure of the NWs right before and right after fracture. The much larger critical strain observed in the thinner NW is related to the

Figure 8.6: Snapshots of Si NWs of two diameters under bending deformation before and after fracture. While metastable hillocks form on the thinner NWs before fracture (a), this does not happen for the thicker NW (c).

formation of metastable hillocks on the compressible side of the NW, as shown in Fig. 8.6(a). It seems that the formation of hillocks relieves some bending strain and allows the thinner NW to deform further without causing fracture. In fact, the onset of hillock formation in the thinner NW happens at the same rotation angle ($\Theta = 0.76$ rad) as the angle at which the thicker NW fractures. The detailed mechanisms responsible for the size and possible temperature dependence of NW failure modes in bending will be presented in a subsequent paper.

8.5 Summary

In this chapter we have presented a unified approach to handle torsion and bending of wires in atomistic simulations by generalizing the Born-von Kármán periodic boundary conditions to cylindrical coordinates. We provided the expressions for the torque and bending moments in terms of an average over the entire simulation cell, in close analogy to the Virial stress expression. Molecular Dynamics simulations under these new boundary conditions show several failure modes of Silicon nanowires under torsion and bending, depending on the nanowire diameter. These simulations are able to probe the intrinsic behavior of nanowires because the artificial end effects are completely removed.

Chapter 9

Conclusion

The main contributions of this dissertation fall under three broad areas. The first is the formulation of AVI for particle simulations. A detailed linear stability analysis of AVI was performed for a two-spring single mass system and it was shown the set of unstable time steps is dense. However the majority of these instabilities are practically insignificant as millions of time steps would be required before they are detectable. In addition we proposed a basic resonance mechanism by which instabilities can appear in this two-spring system. These stability results were extended to a more realistic system, a molecular dynamics analog created using a 2-D periodic harmonic lattice. By also looking at a finite-element analog, we were able to explain why resonant behavior is observed for AVI in molecular dynamics simulations but not finite-element simulations. To quell the significant instabilities in molecular dynamics simulations, we introduced Langevin dynamics as a stochastic thermostat. The formulation of AVI was extended to work in the Langevin setting and the correctness of this extension was verified with numerical results.

The second area is the construction of a black-box FMM that is applicable to non-oscillatory kernels. We showed how a low-rank approximation of the kernel can be formed by using Chebyshev interpolation. This approximation can be viewed as a black box since it is independent of the functional form of the kernel. Combining this with the FMM tree gives rise to a black-box FMM. To accelerate the method, a technique involving SVD compression was prescribed to expedite the matrix-vector products that are responsible for the computational bottleneck. In addition it was shown how the method can be extended to handle periodic systems. Finally numerical tests were conducted to validate the method.

Finally the third area is the development of torsion and bending PBCs for molecular dynamics simulations of silicon nanowires. In addition expressions for the virial torque and virial bending moment were derived, which provide a measure of the amount of torque and bending moment being applied to the nanowire by the torsion and bending PBCs, respectively. Finally through simulations using these PBCs various failure modes were identified depending on the diameter of the nanowire.

Appendix A

Stability of AVI

A.1 AVI algorithm in the r-RESPA case

The AVI algorithm in the r-RESPA case for three potentials is shown in Algorithm 3. This algorithm is identical to r-RESPA.

A.2 Harmonic approximation of the lattice potential

The exact non-linear interaction potential between two masses connected by a spring is assumed to be given by:

$$V_{\mathrm{s}}(r_1, r_2) = \frac{k}{2}\left(\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} - L\right)^2,$$

where k is the spring constant, $r_1 = (x_1, y_1)$ is the position of the first mass, $r_2 = (x_2, y_2)$ is the position of the second, and L is the equilibrium length of the spring. To derive a harmonic approximation, we first assume small displacements of the masses in the x and y directions. Then the position r_i of each node can be decomposed into an equilibrium position e_i and a displacement $d_i = (u_i, v_i)$. Consequently we define the x-displacement of the spring as $\Delta u = u_2 - u_1$ and the y-displacement as

Algorithm 3 AVI Algorithm in the r-RESPA case for the case of three potentials

```
Input: $x^0$, $v^0$, $h_1$, $r_1$, $r_2$, $n_{\text{steps}}$
Output: $(x^i, v^i)$, for all $i$
$i = 0$
$h_2 = r_1 h_1$
$h_3 = r_2 h_2$
$F = -h_1 \nabla V_1(x^0) - h_2 \nabla V_2(x^0) - h_3 \nabla V_3(x^0)$
for $n = 1$ to $n_{\text{steps}}$ do {Outer most loop}
  for $j = 1$ to $r_2$ do {Loop for potential $V_2$}
    for $k = 1$ to $r_1$ do {Loop for potential $V_1$}
      for all $a$ do {Half-kick}
        $v_a^{i+1} = v_a^i + \frac{1}{2} \frac{F_a}{m_a}$
      end for
      $x^{i+1} = x^i + h_1 \, v^{i+1}$ {Drift}
      $F = -h_1 \nabla V_1(x^{i+1})$
      if $k == r_1$ then
        $F = F - h_2 \nabla V_2(x^{i+1})$
      end if
      if $j == r_2$ then
        $F = F - h_3 \nabla V_3(x^{i+1})$
      end if
      for all $a$ do {Half-kick}
        $v_a^{i+1} = v_a^{i+1} + \frac{1}{2} \frac{F_a}{m_a}$
      end for
      $i = i + 1$
    end for
  end for
end for
```

$\Delta v = v_2 - v_1$. V_{s} can now be rewritten as a function of Δu and Δv:

$$V_{\mathrm{s}}(\Delta u, \Delta v) = \frac{k}{2}\left(\sqrt{(\Delta u + L_x)^2 + (\Delta v + L_y)^2} - L\right)^2$$

with $L_x = L\cos\theta$ and $L_y = L\sin\theta$ denoting the equilibrium lengths of the spring in the x and y directions, respectively, and θ representing the angle the spring forms with the (1,0) direction. To approximate V_{s}, we find the Taylor expansion of V_{s} about $(\Delta u, \Delta v) = (0, 0)$. The resulting harmonic potential is

$$V_{\mathrm{s}}^{\mathrm{h}}(\Delta u, \Delta v) = \frac{k}{2}(\Delta u\cos\theta + \Delta v\sin\theta)^2.$$

Now we are ready to compute the stiffness matrix $\mathbf{K}$ for the 2-D triangular harmonic lattice using the harmonic potential $V_{\mathrm{s}}^{\mathrm{h}}$. We start by looking at one row of this matrix. For a given mass m_0 in the lattice, it interacts with six neighboring masses $m_1, \ldots, m_6$. These masses will be labeled in a counter-clockwise manner beginning with 1 for the mass located in the (1,0) direction. As a result $\theta_i = (i-1)\pi/6$ for $i = 1, \ldots, 6$. Then the six harmonic potentials are

$$\begin{aligned}
V_1 &= \frac{k}{2}(u_1 - u_0)^2 & V_4 &= \frac{k}{2}(u_4 - u_0)^2 \\
V_2 &= \frac{k}{2}\left[\frac{1}{2}(u_2 - u_0) + \frac{\sqrt{3}}{2}(v_2 - v_0)\right]^2 & V_5 &= \frac{k}{2}\left[-\frac{1}{2}(u_5 - u_0) - \frac{\sqrt{3}}{2}(v_5 - v_0)\right]^2 \\
V_3 &= \frac{k}{2}\left[-\frac{1}{2}(u_3 - u_0) + \frac{\sqrt{3}}{2}(v_3 - v_0)\right]^2 & V_6 &= \frac{k}{2}\left[\frac{1}{2}(u_6 - u_0) - \frac{\sqrt{3}}{2}(v_6 - v_0)\right]^2
\end{aligned}$$

and the corresponding forces on mass m_0 due to potential V_i in the x and y directions are

$$\begin{aligned}
F_{1x} &= k(u_1 - u_0) & F_{1y} &= 0 \\
F_{2x} &= k\left[\frac{1}{4}(u_2 - u_0) + \frac{\sqrt{3}}{4}(v_2 - v_0)\right] & F_{2y} &= k\left[\frac{\sqrt{3}}{4}(u_2 - u_0) + \frac{3}{4}(v_2 - v_0)\right] \\
F_{3x} &= k\left[\frac{1}{4}(u_3 - u_0) - \frac{\sqrt{3}}{4}(v_3 - v_0)\right] & F_{3y} &= k\left[-\frac{\sqrt{3}}{4}(u_3 - u_0) + \frac{3}{4}(v_3 - v_0)\right] \\
F_{4x} &= k(u_4 - u_0) & F_{4y} &= 0 \\
F_{5x} &= k\left[\frac{1}{4}(u_5 - u_0) + \frac{\sqrt{3}}{4}(v_5 - v_0)\right] & F_{5y} &= k\left[\frac{\sqrt{3}}{4}(u_5 - u_0) + \frac{3}{4}(v_5 - v_0)\right] \\
F_{6x} &= k\left[\frac{1}{4}(u_6 - u_0) - \frac{\sqrt{3}}{4}(v_6 - v_0)\right] & F_{6y} &= k\left[-\frac{\sqrt{3}}{4}(u_6 - u_0) + \frac{3}{4}(v_6 - v_0)\right].
\end{aligned}$$

Using these equations, **K** can be formed row-by-row.

Appendix B

Torsion and Bending of Silicon Nanowires

B.1 Continuum Analogue of Virial Torque Expression

In this appendix, we show that the corresponding expression for the Virial torque given in Eq. (8.21),

$$\tau = \frac{1}{L_z} \left\langle \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} -\frac{\partial V}{\partial (x_i - x_j)} (y_i\, z_i - y_j\, z_j) + \frac{\partial V}{\partial (y_i - y_j)} (x_i\, z_i - x_j\, z_j) \right\rangle$$

is given by Eq. (8.22),

$$\tau = Q_{zz} \equiv \frac{1}{L_z} \int_{\Omega} -y\, \sigma_{xz} + x\, \sigma_{yz}\, dV$$

The reader is referred to the classical text of [29] for a discussion of a related problem. To begin, let us consider a continuum body aligned along the x_j axis and subjected to t-PBC as defined in Section 2.2. The two end surfaces of the continuum are planes perpendicular to the x_j axis: a plane $S^{(a,j)}$ on which $x_j = x_j^{(a)}$, and another plane $S^{(b,j)}$ on which $x_j = x_j^{(b)}$, where $x_j^{(b)} = x_j^{(a)} + L$, as shown in Fig. B.1. The surface normal vectors for both $S^{(a,j)}$ and $S^{(b,j)}$ planes point to the positive x_j-axis. The remaining surface area of the continuum body, S_c, is traction-free. Q_{ij} is defined as the moment around the x_i-axis due to the traction forces acting on these surfaces, which has a surface normal along the x_j-axis. Obviously, the Virial torque τ here should correspond to Q_{zz} because both the wire and the torque are along the z-axis. Hence our task is to show that Q_{zz} can be written into the form of Eq. (8.22).

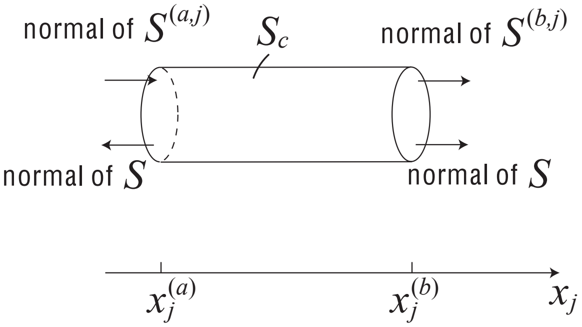


Figure B.1: A continuum body subjected to t-PBC whose surface S consists of three pieces: $-S^{(a,j)}$, $S^{(b,j)}$ and S_c (see text).

We now derive the general expression for Q_{ij}. By definition, we can write Q_{ij} as a surface integral,

$$\begin{aligned} Q_{ij} &\equiv \int_{S^{(a,j)}} (\mathbf{x} \times \mathbf{t})_i \, dS \\ &= \int_{S^{(a,j)}} \epsilon_{ikl} \, x_k \, \sigma_{ml} \, n_m \, dS \end{aligned} \tag{B.1}$$

where $\mathbf{n}$ is the normal vector of the surface, σ_{ml} is the stress field, and we have used the fact that $F_l = \sigma_{ml} \, n_m$. At equilibrium, the moment evaluated at $S^{(a,j)}$ and $S^{(b,j)}$ must equal each other, i.e.,

$$Q_{ij} = \int_{S^{(a,j)}} \epsilon_{ikl} \, x_k \, \sigma_{ml} \, n_m \, dS = \int_{S^{(b,j)}} \epsilon_{ikl} \, x_k \, \sigma_{ml} \, n_m \, dS \tag{B.2}$$

Let us now consider the piece of continuum that is enclosed by the two cross sections, $S^{(a,j)}$ and $S^{(b,j)}$. Let S be the surface area of this continuum that consists of three parts, $-S^{(a,j)}$, $S^{(b,j)}$ and S_c, as shown in Fig. B.1. In other words, the local outward surface normal for S is parallel to the normal vector of $S^{(b,j)}$ but is anti-parallel to the normal vector of $S^{(a,j)}$. We can show that Q_{ij} can be written as an integral over the entire surface S,

$$Q_{ij} \;=\; \frac{1}{x_j^{(b)} - x_j^{(a)}} \oint_S \epsilon_{ikl}\, x_k\, x_j\, \sigma_{ml}\, n_m\, dS \tag{B.3}$$

Notice that the extra term x_j in Eq. (B.3) takes two different constant values on the two surfaces $S^{(b,j)}$ and $S^{(a,j)}$. To show that Eq. (B.3) is equivalent to Eq. (B.2), we used the fact that $x_j = x_j^{(a)}$ on $S^{(a,j)}$, $x_j = x_j^{(b)}$ on $S^{(b,j)}$ and S_c is traction-free. Therefore,

$$\begin{aligned} & \frac{1}{x_j^{(b)} - x_j^{(a)}} \oint_S \epsilon_{ikl}\, x_k\, x_j\, \sigma_{ml}\, n_m\, dS \\ = \;& \frac{1}{x_j^{(b)} - x_j^{(a)}} \left[x_j^{(b)} \int_{S^{(b,j)}} \epsilon_{ikl}\, x_k\, \sigma_{ml}\, n_m\, dS - x_j^{(a)} \int_{S^{(a,j)}} \epsilon_{ikl}\, x_k\, \sigma_{ml}\, n_m\, dS \right] \\ = \;& \int_{S^{(b,j)}} \epsilon_{ikl}\, x_k\, \sigma_{ml}\, n_m\, dS \end{aligned} \tag{B.4}$$

Now that Eq. (B.3) expresses Q_{ij} as an integral over a closed surface, we apply Gauss's Theorem and obtain,

$$\begin{aligned} Q_{ij} \;&=\; \frac{1}{x_j^{(b)} - x_j^{(a)}} \int_\Omega (\epsilon_{ikl}\, x_k\, x_j\, \sigma_{ml})_{,m}\, dV \\ &=\; \frac{1}{x_j^{(b)} - x_j^{(a)}} \int_\Omega \epsilon_{ikl}\, (x_k\, x_j)_{,m}\, \sigma_{ml} \; + \epsilon_{ikl}\, x_k\, x_j\, \sigma_{ml,m}\, dV \end{aligned} \tag{B.5}$$

Given $x_{k,l} = \delta_{kl}$, and the equilibrium condition $\sigma_{ml,l} = 0$, we have

$$\begin{aligned} Q_{ij} \;&=\; \frac{1}{x_j^{(b)} - x_j^{(a)}} \int_\Omega \epsilon_{ikl}\, (\delta_{km}\, x_j + x_k\, \delta_{jm})\, \sigma_{ml}\, dV \\ &=\; \frac{1}{x_j^{(b)} - x_j^{(a)}} \int_\Omega \epsilon_{ikl}\, x_j\, \sigma_{kl} + \epsilon_{ikl}\, x_k\, \sigma_{jl}\, dV \end{aligned} \tag{B.6}$$

Due to the symmetry of the stress tensor, $\epsilon_{ikl}\,\sigma_{kl} = 0$. Hence

$$Q_{ij} \;=\; \frac{1}{x_j^{(b)} - x_j^{(a)}} \int_\Omega \epsilon_{ikl}\, x_k\, \sigma_{jl}\, dV \tag{B.7}$$

In the special case of $x_i = z$ and $x_j = z$, we have

$$T \;=\; Q_{zz} = \frac{1}{L_z} \int_\Omega x\,\sigma_{yz} - y\,\sigma_{xz}\, dV \tag{B.8}$$

We mention in passing that, for a straight rod aligned along the x-axis, the bending moment around the z-axis exerted at its ends can be expressed as,

$$M \;=\; Q_{zx} = \frac{1}{L_x} \int_\Omega x\,\sigma_{yx} - y\,\sigma_{xx}\, dV \tag{B.9}$$

This expression is useful only when the rod is infinitely long and periodic along the x-axis. However, an initially straight rod subject to finite bending moment will become curved (and violates PBC), so that the above equation can no longer be used. In this case, we need to express the bending moment and the tensor Q in cylindrical coordinates (b-PBC) as in the next section.

B.2 Continuum Analogue of Virial Bending Expression

In this appendix, we show that the corresponding expression for Virial bending moment given in Eq. (8.28),

$$\begin{aligned} M \;=\; & \tfrac{1}{\Theta}\Big\langle \textstyle\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} \quad -\frac{\partial V}{\partial(x_i - x_j)}(y_i\,\theta_i - y_j\,\theta_j + \rho\,\cos\theta_i - \rho\,\cos\theta_j) \\ & \qquad\qquad\qquad\qquad + \frac{\partial V}{\partial(y_i - y_j)}(x_i\,\theta_i - x_j\,\theta_j - \rho\,\sin\theta_i + \rho\,\sin\theta_j)\Big\rangle \end{aligned}$$

is given by Eq. (8.30),

$$\begin{aligned} M &= Q_{z\theta} = \frac{1}{\Theta}\int_A dA \int_0^{\Theta} d\theta\,(-y\,\sigma_{x\theta} + x\,\sigma_{y\theta}) \\ &= \frac{1}{\Theta}\int_A dA \int_0^{\Theta} d\theta\; r\,\sigma_{\theta\theta} = \frac{1}{\Theta}\int_\Omega \sigma_{\theta\theta}\,dV \end{aligned}$$

The constructions in the previous section can be generalized to bending by using cylindrical coordinates to show this.

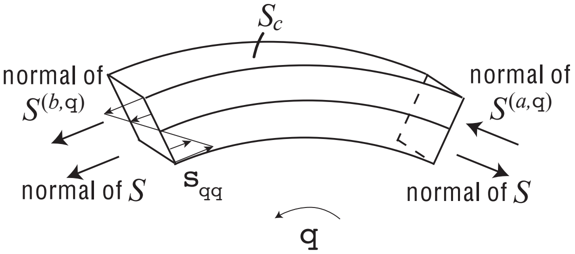


Figure B.2: A continuum body subjected to b-PBC whose surface S consists of three pieces: $-S^{(a,\theta)}$, $S^{(b,\theta)}$ and S_c (see text).

Consider a continuum body aligned along the θ axis and subjected to b-PBC as defined in Section 2.3. The two end surfaces of the continuum are planes perpendicular to the θ axis: plane $S^{(a,\theta)}$ on which $\theta = \theta_a$, and another plane $S^{(b,\theta)}$ on which $\theta = \theta_b$, where $\theta_b = \theta_a + \Theta$, as shown in Fig. B.2. The normal vectors of both planes are in the positive θ direction. The remaining surface area of the continuum body, S_c, is traction-free. To simplify some of the notations, the cross section of the continuum body is assumed to be rectangular. The main conclusion in this section remains valid for a circular cross section. $Q_{z\theta}$ is the bending moment (around z) due to the traction on the half plane $S^{(a,\theta)}$ is,

$$\begin{aligned} Q_{z\theta} &= \int_{S^{(a,\theta)}} (\mathbf{x}\times\mathbf{t})_z\,dS \\ &= \int_{S^{(a,\theta)}} (-y\,\sigma_{xk} + x\,\sigma_{yk})\,n_k\,dS \qquad \text{(B.10)} \end{aligned}$$

At equilibrium, the bending moment evaluated at half-plane $S^{(a,\theta)}$ and $S^{(b,\theta)}$ must be equal because the total moment on the continuum contained inside the wedge between the two half-planes should be zero. Therefore,

$$Q_{z\theta} = \int_{S^{(a,\theta)}} (-y\,\sigma_{xk} + x\,\sigma_{yk})\,n_k\,dS = \int_{S^{(b,\theta)}} (-y\,\sigma_{xk} + x\,\sigma_{yk})\,n_k\,dS \quad \text{(B.11)}$$

Now consider the continuum contained between the two half-planes $S^{(a,\theta)}$ and $S^{(b,\theta)}$. The total surface area of this continuum is S, which consists of $-S^{(a,\theta)}$, $S^{(b,\theta)}$ and S_c (traction free), as shown in Fig. B.2. Similar to the previous section, we can show that

$$Q_{z\theta} = \frac{1}{\theta_b - \theta_a} \oint_S (-y\,\sigma_{xk} + x\,\sigma_{yk})\,\theta\,n_k\,dS \quad \text{(B.12)}$$

Notice the extra term θ in the integrand, which takes different values on the two surfaces $S^{(b,\theta)}$ and $S^{(a,\theta)}$. Applying Gauss's Theorem, we obtain

$$\begin{aligned} Q_{z\theta} &= \frac{1}{\Theta}\int_\Omega (-y\,\theta\,\sigma_{xk} + x\,\theta\,\sigma_{yk})_{,k}\,dV \\ &= \frac{1}{\Theta}\int_\Omega -(y\,\theta)_{,k}\,\sigma_{xk} + (x\,\theta)_{,k}\,\sigma_{yk}\,dV \\ &= \frac{1}{\Theta}\int_\Omega -y\,\theta_{,k}\,\sigma_{xk} + x\,\theta_{,k}\,\sigma_{yk}\,dV \end{aligned} \quad \text{(B.13)}$$

Since $\partial\theta/\partial x = -y/r^2$ and $\partial\theta/\partial y = x/r^2$ we have

$$Q_{z\theta} = \frac{1}{\Theta}\int_\Omega (\sin^2\theta\,\sigma_{xx} + \cos^2\theta\,\sigma_{yy} - 2\sin\theta\cos\theta\,\sigma_{xy})\,dV = \frac{1}{\Theta}\int_\Omega \sigma_{\theta\theta}\,dV \text{(B.14)}$$

Because the final expression, Eq. (B.14), may look counter-intuitive, in the following we will clarify its meaning using an alternative derivation. Indeed, since the wire is not subjected to a net tensile force, we expect the integral of $\sigma_{\theta\theta}$ over any cross section perpendicular to the θ axis to be zero. Hence one might expect the volume integral of $\sigma_{\theta\theta}$ to be zero as well — but this is not the case.

Assume the cross section area of the continuum body (beam) to be a rectangle with height h and width w, i.e. $r \in [\rho - h/2, \rho + h/2]$ and $z \in [-w/2, w/2]$. $r = \rho$

marks the neutral axis of the beam. Define $\xi \equiv r - \rho$. The bending moment on any cross section perpendicular to the θ-axis can be expressed as

$$M = \int_{-h/2}^{h/2} d\xi \int_{-w/2}^{w/2} dz\; \xi\, \sigma_{\theta\theta} \tag{B.15}$$

Since the net tensile force on any cross section should be zero, we have

$$\int_{-h/2}^{h/2} d\xi \int_{-w/2}^{w/2} dz\, \sigma_{\theta\theta} = 0 \tag{B.16}$$

Hence we can re-write M as,

$$\begin{aligned} M &= \int_{-h/2}^{h/2} d\xi \int_{-w/2}^{w/2} dz\, (\xi + \rho)\, \sigma_{\theta\theta} \\ &= \int_{\rho-h/2}^{\rho+h/2} dr \int_{-w/2}^{w/2} dz\, r\, \sigma_{\theta\theta} \end{aligned} \tag{B.17}$$

At mechanical equilibrium, the value of M does not depend on the cross section on which it is calculated. Hence we can also express M as an average over all possible cross sections,

$$M = \frac{1}{\Theta} \int_{\rho-h/2}^{\rho+h/2} dr \int_{-w/2}^{w/2} dz \int_0^{\Theta} d\theta\, r\, \sigma_{\theta\theta} = \frac{1}{\Theta} \int_A dA \int_0^{\Theta} d\theta\, r\, \sigma_{\theta\theta} = \frac{1}{\Theta} \int_\Omega dV \sigma_{\theta\theta} \tag{B.18}$$

B.3 Numerical Verification of the Virial Torque Formula

In this appendix, we provide numerical verification for the Virial torque expression, Eq. (8.21), in the limit of zero temperature. Because the free energy is the same as the potential energy at zero temperature, what this test really verifies is Eq. (8.20) for the derivative of potential energy with respect to twist angle ϕ.

In this test case, we examine a small Si NW described by the Stillinger-Weber

(SW) potential [85]. This is different from the MEAM model we used in the MD simulations presented in Section 4. The reason to use the SW model here is its simplicity since we foresee that some readers may be interested in implementing t-PBC and b-PBC into their own simulation programs. While MD simulations under t-PBC and b-PBC require very little change to the existing programs, the calculation of the Virial torque and bending moment requires modification to the subroutine that computes the Virial stress, which is usually the same subroutine that computes the potential energy and forces on each atom. The modification is simple for simple models such as SW, Lennard-Jones [1], or embedded-atom-method (EAM) [23] potentials, but gets quite complicated for the MEAM model. The test case here provides a benchmark for potential readers who are interested enough to implement the Virial torque expression in the SW model.

We did not use the SW model for MD simulations of Si NW in Section 4 for the following reasons. First, the elastic constants of SW model do not agree well with experimental values. The agreement is worse for the shear modulus than for the Young's modulus. Because the elastic response of a wire under torsion is dictated by its shear modulus, the SW model predicts a very different torque-angle curve from what we would expect from an experiment (and from the predictions of MEAM potential). Second, the SW model (along with many other models) is known to predict an artificially ductile behavior for Si in MD simulations. In comparison, the MEAM model predicts the correct brittle behavior at low temperatures [54]. Nonetheless, the SW model suffices to provide a simple test case to check the self-consistency of Eq. (8.20).

We prepared a NW along the $[1\bar{1}0]$ orientation with diameter $D = 3.2$ nm and length $L_z = 13.8$ nm. The NW was annealed by MD simulation at 1000 K for 1 ps followed by conjugate gradient relaxation to a local energy minimum. After that, we applied a torsion by rotating the two ends of the NW with respect to each other in steps of $\Delta\phi = 0.02$ radian. The atoms were relaxed to their minimum energy positions by conjugate gradient at each twist angle. Next, we took the relaxed configuration for $\phi = 0.1$ and recorded the Virial torque value as $\tau_0 = 15.008629791$ eV. We then added an additional twist $\pm\Delta\phi$ to it and computed the new potential energy $V_\pm$ (without relaxation). The Virial torque can be estimated from the numerical derivative of the potential energy,

$$\tau_{\rm num} = \frac{V_+ - V_-}{2\Delta\phi} \tag{B.19}$$

The values for $\tau_{\rm num}$ and the differences from τ_0 for different values of $\Delta\phi$ are listed below.

$\Delta\phi$	$\tau_{\rm num}(\text{eV})$	$\tau_{\rm num} - \tau_0(\text{eV})$
10^{-1}	15.0081445307	4.85×10^{-4}
10^{-2}	15.0086250291	4.76×10^{-6}
10^{-3}	15.0086297599	3.11×10^{-8}

(B.20)

The numerical differentiation converges to the Virial torque formula at the speed of $\mathcal{O}(\Delta\phi)^2$, as it should. This confirms Eq. (8.20).

B.4 Numerical Verification of the Virial Bending Moment Formula

In this appendix, we provide numerical verification for the Virial bending moment expression, Eq. (8.28), in the limit of zero temperature. Because the free energy is the same as the potential energy at zero temperature, what this test really verifies is Eq. (8.27) for the derivative of potential energy with respect to bending angle Θ. We also used the SW potential for Si here for simplicity.

The NW was created, equilibrated, and relaxed in the same way as described in the previous section. After that we applied bending by rotating the two ends of the NW with respect to each other in steps of $\Delta\Theta = 0.02$ radian. The atoms were relaxed to their minimum energy positions by conjugate gradient at each value of Θ. We took the relaxed configuration for $\Theta = 0.1$ and recorded the Virial bending moment as $M_0 = 30.57105749653973$ eV. We then added an additional twist of $\pm\Delta\Theta$ to it and computed the new potential energy $V_\pm$ (without relaxation). The Virial bending moment can then be estimated from the numerical derivative of the potential energy,

$$M_{\rm num} = \frac{V_+ - V_-}{2\Delta\Theta} \tag{B.21}$$

The values for M_{num} and the differences from M_0 for different values of $\Delta\Theta$ are listed below.

$\Delta\Theta$	$M_{\mathrm{num}}(\mathrm{eV})$	$M_{\mathrm{num}} - M_0(\mathrm{eV})$
5×10^{-2}	30.566664661	4.39×10^{-3}
5×10^{-3}	30.571013585	4.39×10^{-5}
5×10^{-4}	30.571056959	5.38×10^{-7}

(B.22)

The numerical differentiation converges to the Virial bending moment expression at the speed of $\mathcal{O}(\Delta\Theta)^2$, as it should. This confirms Eq. (8.28).

Bibliography

[1] M. P. Allen, D. J. Tildesley, Computer Simulation of Liquids, Oxford University Press, 2007.

[2] B. Alpert, G. Beylkin, R. Coifman, V. Rokhlin, Wavelet-like bases for the fast solution of second-kind integral equations, SIAM J. Sci. Comput. 14 (1) (1993) 159–184.

[3] E. Barth, T. Schlick, Extrapolation versus impulse in multiple-timestepping schemes. II. Linear analysis and applications to Newtonian and Langevin dynamics, J. Chem. Phys. 109 (5) (1998) 1633–1642.

[4] E. Barth, T. Schlick, Overcoming stability limitations in biomolecular dynamics. I. combining force splitting via extrapolation with langevin dynamics in LN, J. Chem. Phys. 109 (5) (1998) 1617–1632.

[5] M. I. Baskes, Modified embedded-atom potentials for cubic materials and impurities, Phys. Rev. B 46 (1992) 2727–2742.

[6] T. Belytschko, Y. Lu, Convergence and stability analyses of multi-time step algorithm for parabolic systems, Computer Methods in Applied Mechanics and Engineering 102 (1993) 179–198.

[7] T. Belytschko, R. Mullen, Mesh partitions of explicit-implicit time integrators, in: K.-J. Bathe, J. Oden, W. Wunderlich (eds.), Formulations and Computational Algorithms in Finite Element Analysis, MIT Press, 1976, pp. 673–690.

[8] G. Benettin, A. Giorgilli, On the Hamiltonian interpolation of near-to-the-identity symplectic mappings with application to symplectic integration algorithms, J. Stat. Phys. 75 (5-6) (1994) 1117–1143.

[9] J. Biesiadecki, R. Skeel, Dangers of multiple time steps methods, Journal of Computational Physics 109 (1993) 318–328.

[10] S. D. Bond, B. J. Leimkuhler, B. B. Laird, The nose-poincare method for constant temperature molecular dynamics, J. Comput. Phys. 151 (1999) 114–134.

[11] V. V. Bulatov, W. Cai, Computer Simulations of Dislocations, Oxford University Press, 2006.

[12] M. Chau, O. Englander, L. Lin, Silicon nanowire-based nanoactuator, in: Proceedings of the 3rd IEEE conference on nanotechnology, vol. 2, San Francisco, CA, 2003.

[13] H. Cheng, Z. Gimbutas, P. G. Martinsson, V. Rokhlin, On the compression of low rank matrices, SIAM J. Sci. Comput. 26 (4) (2005) 1389–1404.

[14] K. S. Chueng, S. Yip, Atomic-level stress in an inhomogeneous system, J. Appl. Phys. 70 (1991) 5688–90.

[15] J. Cormier, J. M. Rickman, T. J. Delph, Stress calculation in atomistic simulations of perfect and imperfect solids, J. Appl. Phys. 89 (2001) 99–104.

[16] Y. Cui, Q. Wei, H. Park, C. M. Lieber, Nanowire nanosensors for highly sensitive and selective detection of biological and chemical species, Science 293 (2001) 1289–1292.

[17] Y. Cui, Z. Zhong, D. Wang, W. U. Wang, C. M. Lieber, High performance silicon nanowire field effect transistors, Nano Letters 3 (2003) 149–152.

[18] W. Dahmen, H. Harbrecht, R. Schneider, Compression techniques for boundary integral equations–asymptotically optimal complexity estimates, SIAM J. Num. Analysis 43 (6) (2006) 2251–2271.

[19] W. J. T. Daniel, Analysis and implementation of a new constant acceleration subcycling algorithm, International Journal for Numerical Methods in Engineering 40 (1997) 2841–2855.

[20] W. J. T. Daniel, The subcycled Newmark algorithm, Computational Mechanics 20 (1997) 272–281.

[21] W. J. T. Daniel, A study of the stability of subcycling algorithms in structural dynamics, Computer Methods in Applied Mechanics and Engineering 156 (1) (1998) 1–13.

[22] W. J. T. Daniel, A partial velocity approach to subcycling structural dynamics, Computer Methods in Applied Mechanics and Engineering 192 (3/4) (2003) 375 – 94.

[23] M. S. Daw, S. M. Foiles, M. I. Baskes, The embedded-atom method: a review of theory and applications, Mat. Sci. Engr. Rep. 9 (1993) 251.

[24] W. Ding, L. Calabri, X. Chen, K. Kohlhass, R. S. Ruoff, Mechanics of crystalline boron nanowires, presented at the 2006 MRS spring meeting, San Francisco, CA (2006).

[25] T. Dumitrica, R. D. James, Objective molecular dynamics, J. Mech. Phys. Solids 55 (2007) 2206–2236.

[26] A. Dutt, M. Gu., V. Rokhlin, Fast algorithms for polynomial interpolation, integration, and differentiation, SIAM J. Num. Analysis 33 (5) (1996) 1689–1711.

[27] A. Dutt, V. Rokhlin, Fast Fourier transforms for nonequispaced data, SIAM J. Sci. Comput. 14 (6) (1993) 1368–1393.

[28] A. Edelman, P. McCorquodale, S. Toledo, The future fast Fourier transform?, SIAM J. Sci. Comput. 20 (3) (1999) 1094–1114.

[29] J. D. Eshelby, The continuum theory of lattice defects, Solid State Physics 3 (1956) 79–144.

[30] F. Ethridge, L. Greengard, A new fast-multipole accelerated poisson solver in two dimensions, SIAM J. Sci. Comput. 23 (3) (2001) 741–760.

[31] R. Fan, R. Karnik, M. Yue, D. Y. Li, A. Majumdar, P. D. Yang, DNA translocation in inorganic nanotubes, Nano Letters 5 (2005) 1633–1637.

[32] Y. Feng, K. Han, D. Owen, Asynchronous/multiple time integrators for impact and discrete systems, Proceedings of 8th International Conference on Computational Plasticity (COMPLAS VIII), Barcelona, Spain (2005) 5–8.

[33] Z. Gimbutas, V. Rokhlin, A generalized fast multipole method for nonoscillatory kernels, SIAM J. Sci. Comput. 24 (3) (2003) 796–817.

[34] A. Gravouil, A. Combescure, Multi-time-step explicit-implicit method for non-linear structural dynamics, International Journal for Numerical Methods in Engineering 50 (1) (2001) 199 – 225.

[35] A. Gravouil, A. Combescure, Multi-time-step and two-scale domain decomposition method for non-linear structural dynamics., International Journal for Numerical Methods in Engineering 58 (10) (2003) 1545 – 69.

[36] L. Greengard, V. Rokhlin, A fast algorithm for particle simulations, J. Comp. Phys. 73 (2) (1987) 325–348.

[37] L. Greengard, V. Rokhlin, A new version of the fast multipole method for the laplace equation in three dimensions, Acta Numerica 6 (1997) 229–270.

[38] H. Grubmüller, H. Heller, A. Windemuth, K. Schulten, Generalized Verlet algorithm for efficient molecular dynamics simulations with long-range interactions, Mol. Sim. 6 (1991) 121–142.

[39] E. Hairer, C. Lubich, The life-span of backward error analysis for numerical integrators, Numerische Mathematik 76 (1997) 441–462.

[40] E. Hairer, C. Lubich, Long-time energy conservation of numerical methods for oscillatory differential equations, SIAM J. Num. Anal. 38 (2) (2001) 414–441.

[41] E. Hairer, C. Lubich, G. Wanner, Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations, Springer Verlag, 2002.

[42] G. Han, Y. Deng, J. Glimm, G. Martyna, Error and timing analysis of multiple time-step integration methods for molecular dynamics, Comp. Phys. Comm. 176 (2007) 271–291.

[43] M. Hochbruck, C. Lubich, A Gautschi-type method for oscillatory second-order differential equations, Num. Math. 83 (3) (1999) 403–426.

[44] M. F. Horstemeyer, J. Lim, W. Y. Lu, D. A. Mosher, M. I. Baskes, V. C. Prantil, S. J. Plimpton, Torsion/simple shear of single crystal copper, J. Eng. Mater. Tech. 124 (2002) 322–328.

[45] Y. Huang, C. M. Lieber, Integrated nanoscale electronics and optoelectronics: Exploring nanoscale science and technology through semiconductor nanowires, Pure Appl. Chem 76 (2004) 2051–2068.

[46] T. Hughes, W. Liu, Implicit-explicit finite elements in transient analysis: Stability theory, Journal of Applied Mechanics 78 (1978) 371–374.

[47] T. Hughes, K. Pister, R. Taylor, Implicit-explicit finite elements in nonlinear transient analysis, Computer Methods In Applied Mechanics And Engineering 17/18 (1979) 159–182.

[48] M. Huhtala, A. Kuronen, K. Kaski, Dynamical simulations of carbon nanotube bending, Int. J. Modern Phys. C 15 (2004) 517–534.

[49] Y. Isono, M. Kiuchi, S. Matsui, Development of electrostatic actuated nano tensile testing device for mechanical and electrical characterstics of FIB deposited carbon nanowire, presented at the 2006 MRS spring meeting, San Francisco, CA (2006).

[50] J. Izaguirre, Q. Ma, T. Matthey, J. Willcock, T. Slabach, B. Moore, G. Viamontes, Overcoming instabilities in Verlet-I/r-RESPA with the mollified impulse method, in: T. Schlick, H. Gan (eds.), Computational Methods for Macromolecules: Challenges and Applications, Proceedings of the 3rd International Workshop on Algorithms for Macromolecular Modeling, Springer Verlag, 2000.

[51] J. Izaguirre, S. Reich, R. Skeel, Longer time steps for molecular dynamics, Journal of Chemical Physics 110 (20) (1999) 9853–9864.

[52] P. M. Jeff, N. A. Fleck, The failure of composite tubes due to combined compression and torsion, J. Mater. Sci. 29 (1994) 3080–3084.

[53] C. Kane, J. Marsden, M. Ortiz, M. West, Variational integrators and the Newmark algorithm for conservative and dissipative mechanical systems, International Journal for Numerical Methods in Engineering 49 (10) (2000) 1295–1325.

[54] K. Kang, W. Cai, Brittle and ductile fracture of semiconductor nanowires – molecular dynamics simulations, Philosophical Magazine 87 (2007) 2169–2189.

[55] T. Kizuka, Y. Takatani, K. Asaka, R. Yoshizaki, Measurements of the atomistic mechanics of single crystalline silicon wires of nanometer width, Phys. Rev. B 72 (2005) 035333–1–6.

[56] B. Leimkuhler, S. Reich, Simulating Hamiltonian Dynamics, Cambridge Monographs on Applied and Computational Mathematics, Cambridge University Press, 2005.

[57] A. Lew, Variational time integrators in computational solid mechanics, Ph.D. thesis, Caltech (2003).

[58] A. Lew, J. Marsden, M. Ortiz, M. West, Asynchronous variational integrators, Archive for Rational Mechanics and Analysis 167 (2) (2003) 85–146.

[59] A. Lew, J. Marsden, M. Ortiz, M. West, Variational time integrators, International Journal for Numerical Methods in Engineering 60 (2004) 153–212.

[60] A. Lew, M. Ortiz, Asynchronous variational integrators in Geometry, Mechanics and Dynamics, volume dedicated to J. Marsden in his 60th birthday, Springer, 2002.

[61] R. MacKay, Some aspects of the dynamics of Hamiltonian systems, in: D. Broomhead, A. Iserles (eds.), The dynamics of numerics and the numerics of dynamics, Clarendon Press, Oxford, 1992, pp. 137–193.

[62] M. A. Makeev, D. Srivastava, Silicon carbide nanowires under external loads: An atomistic simulation study, Phys. Rev. B 74 (2006) 165303.

[63] M. Mandziuk, T. Schlick, Resonance in the dynamics of chemical systems simulated by the implicit- midpoint scheme, Chem. Phys. Lett. 237 (1995) 525.

[64] G. Marc, W. G. McMillian, The virial theorem, Adv. Chem. Phys. 58 (1985) 209–361.

[65] J. Marsden, S. Pekarsky, S. Shkoller, M. West, Variational methods, multisymplectic geometry and continuum mechanics, J. Geometry and Physics 38 (2001) 253–284.

[66] J. Marsden, M. West, Discrete mechanics and variational integrators, Acta Numerica (2001) 357–514.

[67] J. Mason, D. Handscomb, Chebyshev Polynomials, Chapman & Hall/CRC, 2003.

[68] T. Miller III, M. Eleftheriou, P. Pattnaik, A. Ndirango, G. Martyna, Symplectic quaternion scheme for biophysical molecular dynamics, J. Chem. Phys. 116 (2002) 8649.

[69] P. Minary, M. Tuckerman, G. Martyna, Long time molecular dynamics for enhanced conformational sampling in biomolecular systems, Phys. Rev. Lett. 93 (2004) 150201.

[70] S. Müller, M. Ortiz, On the gamma–convergence of discrete dynamics and variational integrators, J. Nonlinear Sci. 14 (2004) 279–296.

[71] K. Mylvaganam, T. Vodenitcharova, L. C. Zhang, The bending-kinking analysis of a single-walled carbon nanotube a combined molecular dynamics and continuum mechanics technique, J. Mater. Sci 41 (2006) 3341–3347.

[72] A. Nakatani, H. Kitagawa, Atomistic study of size effect in torsion tests of nanowire, XXI ICTAM.
URL `http://fluid.ippt.gov.pl/ictam04/text/sessions/docs/MS3/11122/MS3_11122.pd`

[73] M. Neal, T. Belytschko, Explicit-explicit subcycling with non-integer time step ratios for structural dynamic systems, Computers & Structures 6 (1989) 871–880.

[74] T. Nozaki, M. Doyama, Y. Kogure, Computer simulation of high-speed bending deformation in copper, Radiation Effects and Defects in Solids 157 (2002) 217–222.

[75] M. Parrinello, A. Rahman, Polymorphic transitions in single crystals: a new molecular dynamics method, J. Appl. Phys. 52 (1981) 7182–7190.

[76] S. Reich, Backward error analysis for numerical integrators, SIAM Journal on Numerical Analysis 36 (5) (1999) 1549–1570.

[77] A. Sandu, T. Schlick, Masking resonance artifacts in force-splitting methods for biomolecular simulations by extrapolative langevin dynamics, J. Comp. Phys. 151 (1999) 74–113.

[78] J. M. Sanz-Serna, Symplectic integrators for Hamiltonian problems – An overview, Acta numerica (1992) 243–286.

[79] T. Schlick, M. Mandziuk, R. Skeel, K. Srinivas, Nonlinear resonance artifacts in molecular dynamics simulations, J. Comput. Phys. 139 (1998) 1.

[80] R. Skeel, K. Srinivas, Nonlinear stability analysis of area-preserving integrators, SIAM Journal on Numerical Analysis 38 (1) (2000) 129–148.

[81] R. Skeel, G. Zhang, T. Schlick, A family of symplectic integrators: Stability, accuracy, and molecular dynamics applications, SIAM Journal on Scientific Computing 18 (1) (1997) 203–222.

[82] R. D. Skeel, K. Srinivas, Nonlinear stability analysis of area-preserving integrators, SIAM J. Numer. Anal. 38 (1) (2000) 129–148.

[83] P. Smolinski, Y.-S. Wu, An implicit multi-time step integration method for structural dynamics problems, Computational Mechanics.

[84] A. Sommerfeld, Partial Differential Equations in Physics, Lectures on Theoretical Physics, volume VI, Academic Press, 1964.

[85] F. H. Stillinger, T. A. Weber, Computer simulation of local order in condensed phase of silicon, Phys. Rev. B 31 (1985) 5262–5271.

[86] J. S. Stölken, A. G. Evans, A microbend test method for measuring the plasticity length scale, Acta Mater. 46 (1998) 5100–5115.

[87] Y. Suris, Hamiltonian methods of Runge-Kutta type and their variational interpretation, Math. Model. 2 (1990) 78–87.

[88] L. Trefethen, D. Bau III, Numerical Linear Algebra, Society for Industrial and Applied Mathematics, 1997.

[89] D. H. Tsai, Virial theorem and stress calculation in molecular-dynamics, J. Chem. Phys. 70 (1979) 1375–82.

[90] M. Tuckerman, B. Berne, G. Martyna, Reversible multiple time scale molecular dynamics, J. Chem. Phys. 97 (1992) 1990–2001.

[91] D. Wang, Q. Wang, A. Javey, R. Tu, H. Dai, Germanium nanowire field-effect transistors with SiO_2 and high-κ HfO_2, Appl. Phys. Lett. 83 (2003) 2432–2434.

[92] J. M. Wendlandt, J. E. Marsden, Mechanical integrators derived from a discrete variational principle, Physica D 106 (1997) 223–246.

[93] M. West, Variational integrators, Ph.D. thesis, Caltech (2004).

[94] C. Zhang, H. Shen, Buckling and postbuckling analysis of single-walled carbon nanotubes in thermal environments via molecular dynamics simulation, Carbon 44 (2006) 2608–2616.

[95] Y. Zhu, H. D. Espinosa, An electromechanical material testing system for *in situ* electron microscopy and applications, Proc. Nat'l. Acad. Sci. 102 (2005) 14503–14508.

[96] J. A. Zimmerman, E. B. W. III, J. J. Hoyt, R. E. Jones, P. A. Klein, D. J. Bammann, Calculation of stress in atomistic simulation, Modell. Simul. Mater. Sci. Eng. 12 (2004) S319–332.