Chapter 2
Numerical Integrators

At its most basic level, molecular dynamics is about mapping out complicated point sets using trajectories of a system of ordinary differential equations (or, in Chaps. 6–8, a stochastic-differential equation system). The sets are typically defined as the collection of probable states for a certain system. In the case of Hamiltonian dynamics, they are directly associated to a region of the energy landscape. The trajectories are the means by which we efficiently explore the energy surface. In this chapter we address the design of numerical methods to calculate trajectories.

When we use the equations of motion for an atomic system and compute a trajectory, we are producing what we hope to be a representative path of the system. For the moment, we discount any external interactions (such as due to heating at a boundary or other driving forces), so we can think of the system as a closed, Hamiltonian dynamics model. This introduces the *microscopic* perspective which concerns the detailed description of the instantaneous atomic positions and velocities as time is varied. These microscopic paths are the cornerstone of statistical mechanical theory, which is the tool that we will eventually develop for understanding molecular systems at a more abstract level, thus it is essential in developing a computational methodology that we have an understanding of how to generate trajectories reliably and efficiently.

The challenge before us is to compute solutions of

$$\dot{\boldsymbol{q}} = \boldsymbol{M}^{-1}\boldsymbol{p}, \qquad \dot{\boldsymbol{p}} = \boldsymbol{F}(\boldsymbol{q}) = -\nabla U(\boldsymbol{q}),$$

or, more compactly, with $\boldsymbol{z}$ representing the collection of all positions and momenta,

$$\dot{\boldsymbol{z}} = \boldsymbol{f}(\boldsymbol{z}), \qquad \boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{J}\nabla H, \tag{2.1}$$

where $\boldsymbol{J} = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{I} \\ -\boldsymbol{I} & \boldsymbol{0} \end{bmatrix}$, and $H = \boldsymbol{p}^T\boldsymbol{M}^{-1}\boldsymbol{p}/2 + U(\boldsymbol{q})$ is the Hamiltonian.

B. Leimkuhler, C. Matthews, *Molecular Dynamics*, Interdisciplinary Applied Mathematics 39, DOI 10.1007/978-3-319-16375-8_2

In order to correctly model the different possible states of the system, it will be necessary to cover a large part of the accessible phase space, so either trajectories must be very long or we must use many initial conditions. There are many ways to solve initial value problems such as (2.1) combined with an initial condition $z(0) = \boldsymbol{\zeta}$. The methods introduced here all rely on the idea of a discretization with a finite stepsize h, and an iterative procedure that computes, starting from $z_0 = \boldsymbol{\zeta}$, a sequence $z_1, z_2, \ldots$, where $z_n \approx z(nh)$. The simplest scheme is certainly Euler's method which advances the solution from timestep to timestep by the formula[1]

$$z_{n+1} = z_n + h\boldsymbol{f}(z_n).$$

The method is based on the observation that $z(t+h) \approx z(t)+h\dot{z}(t)$, i.e. the beginning of a Taylor series expansion in powers of h, and the further observation that the solution satisfies the differential equation, hence $\dot{z}(t)$ may be replaced by $\boldsymbol{f}(z(t))$.

In order to be able to easily compare the properties of different methods in a unified way, we focus in this chapter primarily on a particular class of schemes, generalized *one-step methods*. Suppose that the system under study has a well defined flow map $\mathscr{F}_t$ defined on the phase space (which is assumed to exclude any singular points of the potential energy function). The solution of the initial value problem, $\dot{z} = \boldsymbol{f}(z)$, $z(0) = \boldsymbol{\zeta}$, may be written $z(t, \boldsymbol{\zeta})$ (with $z(0, \boldsymbol{\zeta}) = \boldsymbol{\zeta}$), and the flow-map $\mathscr{F}_t$ satisfies $\mathscr{F}_t(\boldsymbol{\zeta}) = z(t, \boldsymbol{\zeta})$. A *one-step method*, starting from a given point, approximates a point on the solution trajectory at a given time h units later. Such a method defines a map $\mathscr{G}_h$ of the phase space as illustrated in Fig. 2.1.

We assume here a basic understanding of ordinary differential equations; some good references for review of this topic are [16, 51, 177, 362]. For basic concepts in the numerical analysis of ordinary differential equations the definitive reference is [167].

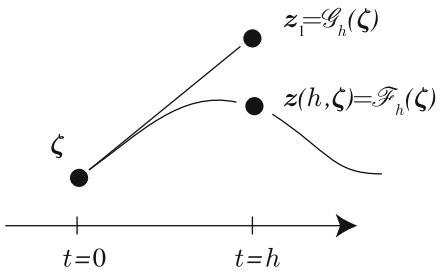

Fig. 2.1 A step with the flow map approximation $\mathscr{G}_h$ is illustrated in comparison to the corresponding step along the solution curve defined by the flow map $\mathscr{F}_h$

[1] Subscripts were used previously to indicate the components of vectors and here they are used to indicate the indexing of timesteps. Although in theory this could lead to some confusion, it normally does not in practice, since we index components in descriptions of details of models and we discuss timesteps in context of defining numerical methods for general classes of systems. Moreover, we use boldface for vectors, so a subscript on a boldface vector indicates a timestep index. When we wish to refer to both the timestep and the component index, we may write $z_{n,i}$ to denote the ith component at timestep n.

Let us emphasize that the issues arising in the design and analysis of numerical methods for molecular dynamics are slightly different than those confronted in other application areas. For one thing the systems involved are highly structured having conservation properties such as first integrals and Hamiltonian structure. We address the issues related to the inherent structure of the molecular N-body problem in both this and the next chapter; wherein we shall learn that *symplectic* discretizations are typically the most appropriate methods.

Another special aspect of the molecular system is that normally it is sensible to use a fixed stepsize, that is each timestep corresponds to a fixed interval of real time. This is in contrast to other applications where it is found to be important to vary the step during simulation. The reason that a uniform stepsize is used is because, in a simulation of many particles, the complexity of the system ensures that if a strong local force is encountered in some corner of the system, a force of similar magnitude will be found somewhere else at the next instant. Even if, occasionally, an instantaneous event is observed that could be controlled by reducing the stepsize, the necessary adaptive machinery can impair the geometric properties and reduce the efficiency of the numerical procedure.[2] There is no trivial way of selecting the stepsize a priori. In typical practice, one performs several trial runs, examining the fluctuations in energy or other easily computable quantities and makes the choice of stepsize in order to keep these within a tolerable range. The molecular dynamics timesteps are typically quite small, measured in femtoseconds, in order to capture the rapid fluctuations of the atoms (in Chap. 4, we discuss ways of increasing the timestep).

2.1 Error Growth in Numerical Integration of Differential Equations

In this section we discuss the issues of convergence and accuracy in numerical integration methods for solving ordinary differential equations.

Let us begin by considering Euler's method in a bit more detail to understand its convergence order. The *convergence* of a numerical method refers to the ability of the method to provide an arbitrary level of accuracy by using small enough timesteps. The *order of accuracy* is the exponent in the power law by which the error in the method is related to the stepsize. For example, when we say that a method is third order accurate, we mean that the global error (on a fixed finite time interval) can be bounded by Kh^3, where h is a sufficiently small timestep and K is a number which depends on the length of the time interval and the features of the problem, but which is independent of h.

The flow map approximation in the case of Euler's method is

[2] A variable stepsize is only used where one expects extreme changes in particle velocities over the course of a simulation (see e.g. [75, 390] for examples arising in radiation damage cascades).

$$\mathscr{G}_h(z) = z + hf(z),$$

and we have

$$z_{n+1} = \mathscr{G}_h(z_n), \qquad n = 0, 1, 2, \ldots \tag{2.2}$$

with $z_0 = \zeta$. The points $\{z_n\}$ are intended to be approximations to the values of the solution. The obvious question is this: how good an approximation is z_n of $z(nh)$?

Let the approximate solution vectors at successive timesteps be $z_0, z_1, \ldots, z_\nu$ where $\nu h = \tau$. We assume that τ, the length of the time interval, is fixed, and ν is an integer parameter representing the total number of timesteps. In order to improve the quality of the approximation, the parameter ν may be increased, as the stepsize is proportionately decreased. The error at step n is defined by $e_n = \|z_n - z(t_n)\|$, where $t_n = nh$; it clearly depends on h. With these definitions one can prove the following result, which is typical of the sorts of results that are available for numerical methods for initial value problems:

Theorem 2.1 *Let $\mathcal{D}$ be a bounded, open region in $\mathbb{R}^m$ such that $f : \mathcal{D} \to \mathbb{R}^m$ is continuously differentiable. Let ζ be an interior point of $\mathcal{D}$ and suppose the initial value problem (2.1) has a unique solution that remains in $\mathcal{D}$ for $t \in [0, \tau]$. Then there exists a constant $C(\tau) > 0$ such that for sufficiently large $\nu \in \mathbb{N}$ the numerical solution z_n remains in $\mathcal{D}$ for $n = 0, 1, \ldots, \nu$, where $h\nu = \tau$, and, moreover, the maximum global error in Euler's method satisfies*

$$\bar{e} := \max_{0 \le n \le \nu} e_n \le C(\tau)h.$$

In short, the error in the approximation obtained on $[0, \tau]$ is reduced in direct proportion to the number of steps taken to cover this interval. Another way to say this is that $\bar{e} \propto h$, or, using the order notation, $\bar{e} = \mathcal{O}(h)$. Because the global error is of order h^r, where $r = 1$, we say that Euler's method is a *first order* method, or that it *converges with order* $r = 1$.

2.1.1 Example: The Trimer

Let us test Euler's method on the trimer model (formulated as a system in $\mathbb{R}^4$). Fixing initial values ($q_0 = (0.5, 1.0); p_0 = (0.1, 0.1)$) we solve the trimer on $[0, 4]$ using the Euler method first with 40 time-steps of length 0.1, next using 80 time-steps of length 0.05 and then with 160 steps of size 0.025, etc. Each calculation results in a different "discrete trajectory." The solutions are graphed in the xy projection in the left panel of Fig. 2.2 with line segments connecting the points; these piecewise linear paths take on the appearance of smooth curves as the stepsize is reduced.

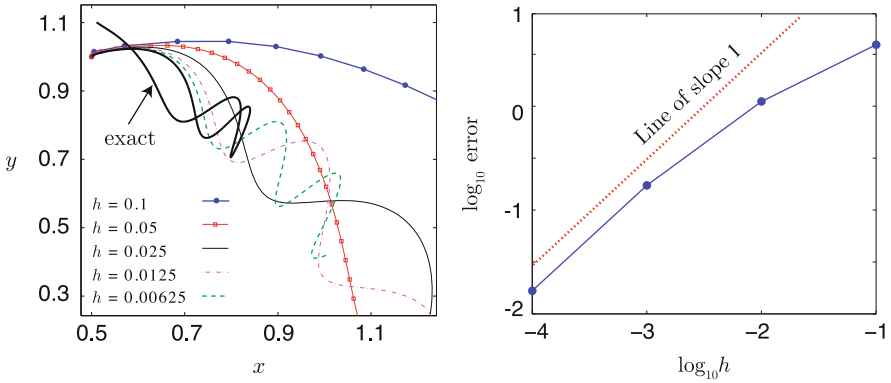

Fig. 2.2 Euler's method applied to the trimer. *Left*, the solution trajectories for different step-sizes; *right*: the maximum global errors in position projections of solutions computed using four different time-steps appear to show an asymptotic linear relationship to step-size, when plotted in log-log scale

A natural question is how can one measure the error in the discrete approximations, since for the trimer, no exact analytical solution is available? Although it is not possible to compute this error exactly, if it is assumed that the process is convergent (as certainly appears to be the case from our experiment) we may use the accurate solution computed with very small steps as a reference and compute the differences between iterates along the trajectories and corresponding points on the reference trajectory. The maximum of the differences between positions of corresponding points is then used as a measure of the global error in the solution for a given step-size h. For the reference solution in the case of the trimer, we have used a simulation with a superior numerical method (the Verlet method) and very small steps of size 10^{-7}. This is the "exact" solution that has been plotted in the left panel of Fig. 2.2. In the right hand panel of Fig. 2.2, the global error $\bar{e}(h)$ is calculated in this way and plotted against the stepsize, using a logarithmic scale. Studying this figure, we see that in log-log scale, the observed relationship between $\bar{e}$ and h is linear, with slope 1 (for h sufficiently small):

$$\ln \bar{e} \approx \ln h + \alpha,$$

where α is a constant. Hence, by exponentiating both sides we obtain

$$\bar{e} \approx Ch,$$

where $C = e^{\alpha}$, confirming the first order relationship between the global error and the step-size.

In the simulations of the example above, it is apparent that the errors are larger at the end of the interval than at earlier times. We know that molecular dynamics trajectories need to be very long compared to the time-step used in

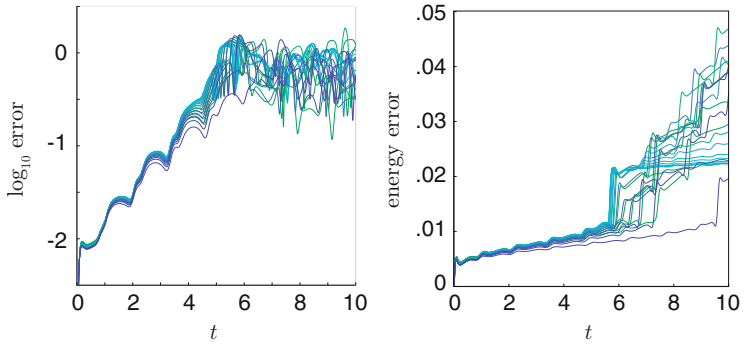

Fig. 2.3 These graphs show the way in which the error grows in relation to time in simulations of the trimer using Euler's method. *Left*: the growth of the logarithm of the trajectory error; *right*: the energy error growth as a function of time

order for them to be useful, so how the error grows in long simulations is quite important. Convergence theorems for numerical methods like that mentioned above are normally formulated for computations on a finite interval, and often do not tell us much about how the error depends on time (i.e. on the length of the interval, τ).

To examine the issue of error growth we perform a simulation using a particular timestep ($h = 0.001$) on the time interval $[0, 10]$ and compare against the reference solution on this interval. The logarithm of the error at each time-step, e_n, is then computed and the result plotted against n. We repeated this for 20 separate initial conditions having the same initial energy. In the left panel of Fig. 2.3, the error graphs are shown, for each of these numerical trajectories, demonstrating that the error in each case grows very rapidly in the early going. Eventually the error growth tapers off, but only when the error is around the same size as the diameter of the system, thus there is little meaningful information remaining regarding a particular trajectory after $t \approx 5$. The right panel shows the errors in energies seen in the same set of simulations, showing that the energy errors initially grow only linearly while the global error is rising exponentially.

To summarize, in this example, with the time interval fixed, the error in Euler's method increases in proportion to the stepsize. On the other hand, the global error grows with the length of the time interval, and at a rapid rate, until it is clear that the numerical trajectory is entirely unrelated to the exact trajectory. Moreover, we observe that the energy errors grow much more slowly than the trajectory errors.

2.1.2 Higher Order Methods

One approach to higher accuracy is to decrease the step-size while continuing to use Euler's method. Since we know that the error on a given fixed interval is proportional to h, using a smaller stepsize should decrease the error in proportion, albeit at the

cost of requiring more timesteps to cover the given time interval. A more efficient means to get better accuracy is to use a higher order method. Higher order methods are ones that satisfy a global error estimate (for finite time intervals) of the form

$$\bar{e} \approx C(\tau)h^r,$$

where $r > 1$.

Example 2.1 The Taylor series expansion of the solution may be written $z(t+h) = z(t) + h\dot{z}(t) + (h^2/2)\ddot{z}(t) + \ldots$. Whereas a first order truncation of this series leads to Euler's method, retaining terms through second order leads to

$$z_{n+1} = z_n + h\dot{z}_n + \frac{h^2}{2}\ddot{z}_n,$$

which is referred to as the 2nd order Taylor series method. In this formula $\dot{z}_n = \boldsymbol{f}(z_n)$, and the second derivative is obtained by differentiating the differential equation itself:

$$\ddot{z}(t) = \frac{\mathrm{d}}{\mathrm{d}t}\dot{z}(t) = \frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{f}(z(t)) = \boldsymbol{f}'(z(t))\dot{z}(t) = \boldsymbol{f}'(z(t))\boldsymbol{f}(z(t)),$$

so one may write the 2nd order Taylor series method as

$$z_{n+1} = z_n + h\boldsymbol{f}(z_n) + \frac{h^2}{2}\boldsymbol{f}'(z_n)\boldsymbol{f}(z_n).$$

This method generates the flow map approximation

$$\mathscr{G}_h(z) = z + h\boldsymbol{f}(z) + \frac{h^2}{2}\boldsymbol{f}'(z)\boldsymbol{f}(z).$$

Note that by the notation $\boldsymbol{f}'(z)$ where $z \in \mathbb{R}^m$ and $\boldsymbol{f} : \mathbb{R}^m \to \mathbb{R}^m$, is meant the $m \times m$ Jacobian matrix whose ij-component is $(\boldsymbol{f}'(z))_{ij} = \partial f_i/\partial z_j$. An alternative notation for $\boldsymbol{f}'$ is $\partial \boldsymbol{f}/\partial z$.

Methods like the Taylor series method offer the prospect of better accuracy in the local approximation, and smaller global error in a given simulation, but they do not necessarily resolve the more important issue relevant for very long time integrations (which we will need in molecular dynamics): the unlimited growth of perturbations from the energy surface. In molecular dynamics, we have already seen that the Euler method has growing energy error which suggests that it will be a poor scheme where the goal is to approximate the behavior of a constant energy trajectory. This same qualitative behavior is seen in other numerical methods, such as the Taylor series method. However, there are alternatives that give both higher order of accuracy and, typically, improved energy accuracy. We discuss one of the most popular schemes of this type in the next section.

2.2 The Verlet Method

The Verlet method (also known as leapfrog or Störmer-Verlet) is a second order method that is popular for molecular simulation. It is specialized to problems that can be expressed in the form $\dot{\boldsymbol{q}} = \boldsymbol{v}, \boldsymbol{M}\dot{\boldsymbol{v}} = \boldsymbol{F}(\boldsymbol{q})$, with even dimensional phase space $\mathbb{R}^{2N}$, which includes constant energy molecular dynamics. Some generalizations exist for other classes of Hamiltonian systems.

The Verlet method is a numerical method that respects certain conservation principles associated to the continuous time ordinary differential equations, i.e. it is a geometric integrator. Maintaining these conservation properties is essential in molecular simulation as they play a key role in maintaining the physical environment. As a prelude to a more general discussion of this topic, we demonstrate here that it is possible to derive the Verlet method from the variational principle. This is not the case for every convergent numerical method. The Verlet method is thus a special type of numerical method that provides a discrete model for classical mechanics.

2.2.1 Hamilton's Principle and Equations of Motion

Hamilton's principle of least action provides a mechanism for deriving equations of motion from a Lagrangian. Recall from Chap. 1 that the Lagrangian for the N-body system is defined by

$$L(\boldsymbol{q}, \boldsymbol{v}) \stackrel{\text{def}}{=} \frac{\boldsymbol{v}^T \boldsymbol{M} \boldsymbol{v}}{2} - U(\boldsymbol{q}),$$

where $\boldsymbol{M}$ is the mass matrix and the potential energy function U is, for simplicity, taken to be smooth (C^2). We consider the collection of all twice continuously differentiable curves in the configuration space which start from a certain point $\boldsymbol{Q}$ and end at another given point $\boldsymbol{Q}'$. We may think of any such curve as being represented by a parameterization $\boldsymbol{q}(t)$, $t \in [\alpha, \beta]$ with $\boldsymbol{q}(\alpha) = \boldsymbol{Q}$ and $\boldsymbol{q}(\beta) = \boldsymbol{Q}'$, where the components of $\boldsymbol{q}(t)$ are C^∞ functions. Denote by $G = G(\boldsymbol{Q}, \boldsymbol{Q}', \alpha, \beta)$ the class of smooth parameterized curves such that $\boldsymbol{q}(\alpha) = \boldsymbol{Q}, \boldsymbol{q}(\beta) = \boldsymbol{Q}'$. Then the *classical action* (or, simply, *action*) of the Lagrangian L is defined for any $\Gamma \in G$ by

$$\mathcal{A}_L(\Gamma) \stackrel{\text{def}}{=} \int_\alpha^\beta L(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t)) \mathrm{d}t.$$

The variational calculus approach to classical mechanics is based on minimizing the action $\mathcal{A}_L$ over the class G of parameterized curves. This is normally referred to as the "principle of least action". It is difficult to provide a physical motivation for this concept, but it is normally taken as a foundation stone for classical mechanics.

Given Γ in G with parameterization $\boldsymbol{q}(t)$, $t \in [\alpha, \beta]$, we consider the curve Γ^ε with parameterization $\boldsymbol{q}^\varepsilon$ defined by

$$\boldsymbol{q}^\varepsilon(t) = \boldsymbol{q}(t) + \varepsilon\boldsymbol{\eta}(t), \qquad t \in [\alpha, \beta], \tag{2.3}$$

where $\boldsymbol{\eta}(t)$ satisfies $\boldsymbol{\eta}(\alpha) = \boldsymbol{\eta}(\beta) = 0$. Thus $\boldsymbol{\eta}$ is a C^∞ parameterized curve linking $\mathbf{0}$ to $\mathbf{0}$. In defining variations in this way, we are using, implicitly, the fact that we can add together functions in C^∞ and multiply them by scalars (e.g. ε) and remain in C^∞. We also make use of the intuitive concept that (2.3) defines $\boldsymbol{q}^\varepsilon$ in such a way that it is "close" to $\boldsymbol{q}$ when ε is small. This can be made precise by a little more elaboration, but we forego this here. Effectively, we are using our understanding of C^∞ as a *normed function space* to restrict attention to variations of the base curve Γ in a particular "direction."

Using a Taylor series expansion of the Lagrangian,[3] we have

$$\begin{aligned}\mathcal{A}_L(\Gamma^\varepsilon) - \mathcal{A}_L(\Gamma) &= \int_\alpha^\beta [L(\boldsymbol{q}(t) + \varepsilon\boldsymbol{\eta}(t), \dot{\boldsymbol{q}}(t) + \varepsilon\dot{\boldsymbol{\eta}}(t)) - L(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t))]\,\mathrm{d}t,\\ &= \int_\alpha^\beta \left[\varepsilon\left(\frac{\partial L}{\partial \boldsymbol{q}}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t))\boldsymbol{\eta}(t) + \frac{\partial L}{\partial \dot{\boldsymbol{q}}}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t))\dot{\boldsymbol{\eta}}(t)\right) + \mathcal{O}(\varepsilon^2)\right]\mathrm{d}t.\end{aligned}$$

Hamilton's principle states that the natural motion of the system described by the Lagrangian L is a stationary point of the classical action which implies that the $\mathcal{O}(\varepsilon)$ term above should vanish for any smooth variation $\boldsymbol{\eta}(t)$ with $\boldsymbol{\eta}(\alpha) = \boldsymbol{\eta}(\beta) = 0$, i.e.,

[3] In the multidimensional setting, Taylor's theorem states that given a C^{k+1} function $f : \mathbb{R}^m \to \mathbb{R}$ and a point z_0, we have

$$\begin{aligned}f(z) - f(z_0) = \nabla f(z_0)\cdot(z - z_0) + f^{(2)}\langle z - z_0, z - z_0\rangle + f^{(3)}\langle z - z_0, z - z_0, z - z_0\rangle\\ + \ldots f^{(k)}\langle z - z_0, z - z_0, \ldots, z - z_0\rangle + \mathcal{O}(\|z - z_0\|^{k+1})\end{aligned}$$

where $\nabla f = \partial f/\partial z$ is the gradient, i.e. a vector with m components, $f^{(2)}$ is the $m \times m$ Hessian matrix of f (the matrix whose ij component is $\partial^2 f/\partial z_i \partial z_j$), and $f^{(2)}\langle \boldsymbol{u}, \boldsymbol{v}\rangle$, $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^m$, represents the quadratic form $\boldsymbol{u}^T f^{(2)}\boldsymbol{v}$. In a similar way we interpret $f^{(3)}$ as a tensor which we can think of as a $m \times m \times m$ triply-indexed array, the ijk element being $\partial^3 f/\partial z_i \partial z_j \partial z_k$ and

$$f^{(3)}\langle \boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}\rangle = \sum_{i=1}^m \sum_{j=1}^m \sum_{k=1}^m (\partial^3 f/\partial z_i \partial z_j \partial z_k) u_i v_j w_k.$$

$$I = \int_{\alpha}^{\beta} \left[\frac{\partial L}{\partial \boldsymbol{q}}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t))\boldsymbol{\eta}(t) + \frac{\partial L}{\partial \dot{\boldsymbol{q}}}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t))\dot{\boldsymbol{\eta}}(t) \right] \mathrm{d}t = 0.$$

We use integration by parts to remove the differentiation of $\boldsymbol{\eta}$, thus (in light of the boundary conditions on $\boldsymbol{\eta}(t)$),

$$I = \int_{\alpha}^{\beta} \left[\frac{\partial L}{\partial \boldsymbol{q}}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t)) - \frac{\mathrm{d}}{\mathrm{d}t} \frac{\partial L}{\partial \dot{\boldsymbol{q}}}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t)) \right] \boldsymbol{\eta}(t) \mathrm{d}t = 0.$$

Since the variation $\boldsymbol{\eta}$ is meant to be arbitrary, it requires

$$\frac{\mathrm{d}}{\mathrm{d}t} \frac{\partial L}{\partial \dot{\boldsymbol{q}}}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t)) = \frac{\partial L}{\partial \boldsymbol{q}}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t)),$$

which is precisely the Lagrangian formulation of the equations of motion.

The method of derivation described here may be formulated in direct analogy to the traditional method of minimizing a function of several variables based on finding critical points. Define the *variational derivative* of a functional $\mathcal{F}$, $\delta\mathcal{F}/\delta\boldsymbol{q}$ so that

$$\mathcal{F}(\boldsymbol{q} + \varepsilon\boldsymbol{\eta}) = \varepsilon \frac{\delta \mathcal{F}}{\delta \boldsymbol{q}} \boldsymbol{\eta} + O(\varepsilon^2),$$

for all suitable (say, C^∞) functions $\boldsymbol{\eta}$. Viewing the action $\mathcal{A}_L(\Gamma)$ as the functional $\mathcal{F}$, the stationarity condition (the Euler-Lagrange equations) may be written

$$\frac{\delta \mathcal{F}}{\delta \boldsymbol{q}} = 0.$$

Thus the calculus of variations becomes a generalization of the traditional method of minimizing smooth functions by finding critical points.[4]

[4]This discussion is a great simplification. Any curve which satisfies this equation will represent a "stationary point" (actually, "stationary curve" would be more accurate) of the classical action. Such curves could include smooth local action minimizers, local action maximizers, or "saddle points" of the actional functional in a generalized sense. Deciding whether a given stationary curve is an actual minimizer of the action would require analysis of the second variation (the coefficient of ε^2 in the expansion above), which introduces additional complexity. For a more comprehensive treatment, see e.g. [210].

2.2.2 Derivation of the Verlet Method

Now let us consider the discrete version of "minimizing the action". We work on the time interval $[0, \tau]$. Consider the $\nu + 1$ points $\boldsymbol{q}_0$ to $\boldsymbol{q}_\nu$ in configuration space

$$\hat{\Gamma} = (\boldsymbol{q}_0, \boldsymbol{q}_1, \ldots, \boldsymbol{q}_\nu).$$

We must first define the action for such a discrete path. One perspective is that we first formulate a discrete version of the Lagrangian in terms of only the point set $\boldsymbol{q}_0, \boldsymbol{q}_1, \ldots \boldsymbol{q}_\nu$, but this somehow requires that we define velocities. Noting that

$$\dot{\boldsymbol{q}}(t) \approx \frac{\boldsymbol{q}(t+h) - \boldsymbol{q}(t)}{h},$$

we are led to consider the approximation at time level n

$$\boldsymbol{v}_n \stackrel{\text{def}}{=} \frac{\boldsymbol{q}_{n+1} - \boldsymbol{q}_n}{h},$$

and thus, by Riemann summation

$$\mathcal{A}_L \approx \hat{\mathcal{A}} \stackrel{\text{def}}{=} \sum_{n=0}^{\nu-1} L\left(\boldsymbol{q}_n, \frac{\boldsymbol{q}_{n+1} - \boldsymbol{q}_n}{h}\right) h.$$

At first glance, it looks like this is a crude approximation, since we have employed a one-sided difference, however, let us proceed anyway to see the implications of this choice. In the case of a mechanical system with Lagrangian $L(\boldsymbol{q}, \boldsymbol{v}) = \dot{\boldsymbol{v}}^T \boldsymbol{M} \boldsymbol{v}/2 + U(\boldsymbol{q})$, we then have

$$\hat{\mathcal{A}} = \sum_{n=0}^{\nu-1} \left[\frac{(\boldsymbol{q}_{n+1} - \boldsymbol{q}_n)^T \boldsymbol{M} (\boldsymbol{q}_{n+1} - \boldsymbol{q}_n)}{2h^2} - U(\boldsymbol{q}_n) \right] h.$$

We should think of $\hat{\mathcal{A}}$ as a function of all the positions $\boldsymbol{q}_0, \boldsymbol{q}_1, \ldots, \boldsymbol{q}_\nu$ defining the discrete path. Critical points of this function satisfy

$$\nabla \hat{\mathcal{A}} = 0,$$

where the gradient must be taken with respect to all configurational points on the path (and all coordinates). This condition leads to the equations

$$\frac{\partial \hat{\mathcal{A}}}{\partial \boldsymbol{q}_n} = 0, \qquad n = 1, \ldots, \nu,$$

since we think of the starting point $\boldsymbol{q}_0$ as fixed. To calculate the derivative of $\hat{\mathcal{A}}$ with respect to $\boldsymbol{q}_n$, $n = 1, 2, \ldots, \nu - 1$, note that only a few terms of the discrete action involve this configurational point, thus

$$\begin{aligned}\frac{\partial \hat{\mathcal{A}}}{\partial \boldsymbol{q}_n} &= \frac{\partial}{\partial \boldsymbol{q}_n}\left[\frac{1}{2h}\left((\boldsymbol{q}_n - \boldsymbol{q}_{n-1})^T \boldsymbol{M}(\boldsymbol{q}_n - \boldsymbol{q}_{n-1}) + (\boldsymbol{q}_{n+1} - \boldsymbol{q}_n)^T\right.\right.\\ &\qquad \left.\left. \times\ \boldsymbol{M}(\boldsymbol{q}_{n+1} - \boldsymbol{q}_n)\right) - U(\boldsymbol{q}_n)h\right]\\ &= \frac{1}{h}\left[\boldsymbol{M}(\boldsymbol{q}_n - \boldsymbol{q}_{n-1}) - M(\boldsymbol{q}_{n+1} - \boldsymbol{q}_n)\right] - \nabla U(\boldsymbol{q}_n)h,\end{aligned}$$

which yields the equations

$$\boldsymbol{M}(\boldsymbol{q}_{n+1} - 2\boldsymbol{q}_n + \boldsymbol{q}_{n-1}) = -h^2 \nabla U(\boldsymbol{q}_n), \qquad n = 1, 2, \ldots, \nu - 1. \tag{2.4}$$

We can think of

$$\frac{1}{h^2}(\boldsymbol{q}_{n+1} - 2\boldsymbol{q}_n + \boldsymbol{q}_{n-1})$$

as a centered finite difference approximation of the acceleration (the second derivative of position), thus the Eq. (2.4) is a direct discretization of Newton's equations.

Somewhat surprisingly, the one-sided approximation of velocities in the discrete Lagrangian has led to a symmetric discretization of the equations of motion. There is a slight issue of what to do at the endpoints. In the variational formulation we think of the endpoints of the curve as fixed points which means that we are effectively solving a boundary value problem and Eq. (2.4) tells us how to compute all the interior points along the path. In the case of an initial value problem (the usual issue in molecular dynamics) we do not know in advance the right endpoint value, but we assume that we can, in some way, calculate $\boldsymbol{q}_1$ (by a *starting procedure*) and so Eq. (2.4) defines $\boldsymbol{q}_2, \boldsymbol{q}_3, \ldots, \boldsymbol{q}_\nu$.

The method (2.4) is commonly referred to as *Störmer's rule*. It was used by the mathematician Störmer for calculations in the first decade of the 1900s. In molecular dynamics this method is referred to as the Verlet method since it was used by Verlet in his important 1967 paper [387].

The scheme is usually given in an alternative "velocity Verlet" form that takes a step from a given vector $\boldsymbol{q}_n$, $\boldsymbol{v}_n$ to $\boldsymbol{q}_{n+1}$, $\boldsymbol{v}_{n+1}$ by the sequence of operations

$$\boldsymbol{v}_{n+1/2} = \boldsymbol{v}_n + (h/2)\boldsymbol{M}^{-1}\boldsymbol{F}_n, \tag{2.5}$$

$$\boldsymbol{q}_{n+1} = \boldsymbol{q}_n + h\boldsymbol{v}_{n+1/2}, \tag{2.6}$$

$$\boldsymbol{v}_{n+1} = \boldsymbol{v}_{n+1/2} + (h/2)\boldsymbol{M}^{-1}\boldsymbol{F}_{n+1}, \tag{2.7}$$

where $\boldsymbol{F}_n = \boldsymbol{F}(\boldsymbol{q}_n) = -\nabla U(\boldsymbol{q}_n)$. The force computed at the end of a given step may be reused at the start of the following step, thus effectively a single force evaluation is needed at each timestep (the method has a similar "cost" to the Euler method, if we measure cost in terms of force evaluations). The derivation of the Störmer form from the velocity Verlet form is straightforward: write down two consecutive steps of (2.5)–(2.7) then eliminate velocities.

The method is widely used but is often stated in other forms, so let us consider these here.

The most straightforward rewriting of the Verlet method is to put the equations and the discretization in Hamiltonian form, i.e. introducing momenta $\boldsymbol{p} = \boldsymbol{M}\boldsymbol{v}$, and thus $\boldsymbol{p}_n = \boldsymbol{M}\boldsymbol{v}_n$, which results in the flow map approximation (taking us from any point in phase space $(\boldsymbol{q}, \boldsymbol{p})$ to a new point $(\boldsymbol{Q}, \boldsymbol{P})$):

$$\boldsymbol{Q} = \boldsymbol{q} + h\boldsymbol{M}^{-1}\boldsymbol{p} + \frac{h^2}{2}\boldsymbol{M}^{-1}\boldsymbol{F}(\boldsymbol{q}), \qquad \boldsymbol{P} = \boldsymbol{p} + \frac{h}{2}[\boldsymbol{F}(\boldsymbol{q}) + \boldsymbol{F}(\boldsymbol{Q})]. \tag{2.8}$$

Alternatively,

$$\boldsymbol{p}_{n+1/2} = \boldsymbol{p}_n + (h/2)\boldsymbol{F}_n, \quad \boldsymbol{q}_{n+1} = \boldsymbol{q}_n + h\boldsymbol{M}^{-1}\boldsymbol{p}_{n+1/2}, \quad \boldsymbol{p}_{n+1} = \boldsymbol{p}_{n+1/2} + (h/2)\boldsymbol{F}_{n+1}.$$

Returning to (2.5)–(2.7), write out the formulas for two successive steps ($(q_{n-1}, v_{n-1}) \mapsto (q_n, v_n)$ and $(q_n, v_n) \mapsto (q_{n+1}, v_{n+1})$) and note that, from

$$\boldsymbol{v}_n = \boldsymbol{v}_{n-1/2} + (h/2)\boldsymbol{M}^{-1}\boldsymbol{F}_n,$$

one has

$$\begin{aligned} \boldsymbol{v}_{n+1/2} &= \boldsymbol{v}_{n-1/2} + h\boldsymbol{M}^{-1}\boldsymbol{F}_n, \\ \boldsymbol{q}_{n+1} &= \boldsymbol{q}_n + h\boldsymbol{v}_{n+1/2}. \end{aligned}$$

How do we use this if, as is typical, initial conditions of the form $\boldsymbol{q}(0) = \boldsymbol{q}_0, \boldsymbol{v}(0) = \boldsymbol{v}_0$ are specified? It is necessary to define an initialization procedure for $\boldsymbol{v}_{-1/2}$:

$$\boldsymbol{v}_{-1/2} = \boldsymbol{v}_0 - (h/2)\boldsymbol{M}^{-1}\boldsymbol{F}(\boldsymbol{q}_0).$$

And it is also necessary to use, at any subsequent step,

$$\boldsymbol{v}_n = \boldsymbol{v}_{n-1/2} + (h/2)\boldsymbol{M}^{-1}\boldsymbol{F}_n,$$

if $\boldsymbol{q}_n, \boldsymbol{v}_n$ are both needed, e.g. for the evaluation of the energy or other velocity-dependent observable.

The difference between formulations is typically subtle, and largely influence the details of computer software design. For many applications the difference can

be safely ignored, but it can have implications when used as part of a more complex algorithm (see [29] and the discussion therein). The use of different formulations can result in differences in the accumulation of rounding error [167].[5]

2.2.3 Convergence and the Order of Accuracy

A typical integrator computes successive steps from the formulas

$$z_{n+1} = \mathscr{G}_h(z_n), \qquad z_0 = \boldsymbol{\zeta}.$$

Assume that $\mathscr{G}_h$ is a smooth map for all $h > 0$. The exact solution satisfies

$$z(t_{n+1}) = \mathscr{F}_h(z(t_n)).$$

To each $h > 0$ we may associate a finite set of phase space points $z_0, z_1, z_2, \ldots, z_\nu$; these represent the numerical solution at $t_0 = 0, t_1 = h, t_2 = 2h, \ldots, t_\nu = \nu h = \tau$.

Taking the difference of the numerical and exact solutions, we have

$$z_{n+1} - z(t_{n+1}) = \mathscr{G}_h(z_n) - \mathscr{F}_h(z(t_n)). \tag{2.9}$$

The first assumption is that $\mathscr{G}_h$ is an $\mathcal{O}(h^{p+1})$ approximation of $\mathscr{F}_h$ in the sense that there is a constant $K \geq 0$ and a constant $\Delta > 0$ such that, for $t \in [0, \tau]$, we have

$$\|\mathscr{F}_h(z(t)) - \mathscr{G}_h(z(t))\| \leq \bar{K} h^{p+1}, \qquad h < \Delta. \tag{2.10}$$

This assumption is usually verified by expanding the numerical and exact solutions in powers of h, using Taylor series expansions.

To tackle the question of the growth of local error, we still must make an important assumption on $\mathscr{G}_h$, namely that it satisfies a *Lipschitz condition* of the form

$$\|\mathscr{G}_h(\boldsymbol{u}) - \mathscr{G}_h(\boldsymbol{w})\| \leq (1 + hL)\|\boldsymbol{u} - \boldsymbol{w}\|, \qquad \boldsymbol{u}, \boldsymbol{w} \in D, h \leq \Delta. \tag{2.11}$$

The set D should be a domain containing the exact solution for $[0, \tau]$, and it is assumed that, for all $h \leq \Delta$ the numerical solution is also contained in D for $n =$

[5] Rounding error is the error introduced when numbers are forced into the finite word length representation in a typical digital computer. Adding together two "computer numbers," then rounding, results in another computer number. Rounding errors may accumulate in long computations, but in molecular dynamics they are normally dominated by the much larger "truncation errors" introduced in the process of discretization, that is, due to replacing the differential equation by a difference equation such as the Euler or Verlet method. For an example of the role of rounding error in the context of constrained molecular dynamics, see [237].

$0, 1, \ldots, \nu$. (This is a simplifying assumption the removal of which would require somewhat more intricate assumptions regarding the differential equations.)

With these assumptions in hand, begin from (2.9) and write

$$z_{n+1} - z(t_{n+1}) = \mathscr{G}_h(z_n) - \mathscr{G}_h(z(t_n)) + \mathscr{G}_h(z(t_n)) - \mathscr{F}_h(z(t_n)),$$

then take norms and use the triangle inequality and (2.10), (2.11) to get the following recurrent inequality for the error $\varepsilon_n = \|z_n - z(t_n)\|$:

$$\varepsilon_{n+1} \leq (1 + Lh)\varepsilon_n + \bar{K}h^{p+1}.$$

From this, the bound

$$\varepsilon_n \leq \frac{\bar{K}}{L} e^{Lnh} h^p, \qquad n = 0, 1, \ldots \nu, \tag{2.12}$$

follows by a straightforward calculation, for $h \leq \Delta$.

The assumption (2.10) that the local error is of order $p + 1$, $p > 0$, is termed the *consistency* of the numerical method. We say the method is *consistent of order p*.

The assumption (2.11) that the method does not increase the separation between two nearby trajectories by more than a factor of the form $1 + hL$ in each step is referred to as the *stability* of the method.

This result shows that a method which is consistent of order p and stable is convergent of order p.

Example 2.2 (The Verlet Method is 2nd Order) In this example, assume a single degree of freedom system, i.e. $q, p \in \mathbb{R}$, and take $M = 1$. The Verlet method can be written in the form of a map, as in (2.8), or, in slightly more detail, as

$$Q = q + hp + \frac{h^2}{2} F(q), \tag{2.13}$$

$$P = p + \frac{h}{2}\left[F(q) + F(q + hp + \frac{h^2}{2}F(q))\right]. \tag{2.14}$$

The first equation is already a polynomial, i.e. it is in the form of a series expansion in powers of h where the coefficients are functions of the starting point (q, p). The second equation may be written as a series expansion in powers of h as well:

$$P = p + \frac{h}{2}F(q) + \frac{h}{2}\left[F(q) + hF'(q)(p + \frac{h}{2}F(q)) + \frac{h^2}{2}F''(q)(p + \frac{h}{2}F(q))^2 + \ldots\right].$$

Note that the neglected terms will involve 4th (and higher) powers of h.

Combining terms of like powers of h, we have

$$P = p + hF + \frac{h^2}{2}pF' + \frac{h^3}{4}\left[F'F + p^2F''\right] + \mathcal{O}(h^4).$$

We carry these terms to compare against the expansion of the exact solution. Since $\dot{q} = p$, we have $\ddot{q} = \dot{p} = F(q)$, and the third derivative is $q^{(3)} = F'(q)p$. On the other hand $\dot{p} = F(q)$ implies that $\ddot{p} = F'(q)\dot{q} = pF'$, and thus

$$p^{(3)} = p^2F'' + F'F.$$

The Taylor expansion of the solution is (taking $q(t) = q, p(t) = p$):

$$q(t+h) = q + hp + \frac{h^2}{2}F + \frac{h^3}{6}F'p + \mathcal{O}(h^4),$$
$$p(t+h) = p + hF + \frac{h^2}{2}pF' + \frac{h^3}{6}\left[p^2F'' + F'F\right] + \mathcal{O}(h^4).$$

We now examine the series expansions for the exact and Verlet solutions and find that these differ in the third (and higher) order terms.

$$Q - q(t+h) = \frac{h^3}{6}F'p + \mathcal{O}(h^4),$$

and

$$P - p(t+h) = \frac{h^3}{12}\left[p^2F'' + F'F\right] + \mathcal{O}(h^4).$$

These relations can be summarized as telling us that

$$\|\mathscr{G}_h(\boldsymbol{z}) - \mathscr{F}_h(\boldsymbol{z})\| = \kappa(\boldsymbol{z})h^3 + \mathcal{O}(h^4),$$

where $\kappa(\boldsymbol{z}) = \kappa(q,p)$ is a function of the position and momentum.

We may then define

$$\bar{K} = \max_{t\in[0,\tau]} \kappa(\boldsymbol{z}(t))$$

bounding the local error by (with neglect of the fourth order terms) $\bar{K}h^3$. Thus the Verlet method is consistent of order two.

To complete the convergence proof for Verlet's method, we would still need to verify the second assumption. This requires the assumption that the force field $\boldsymbol{F}$ satisfy a Lipschitz condition:

$$\|\boldsymbol{F}(\boldsymbol{u}) - \boldsymbol{F}(\boldsymbol{w})\| \leq \hat{L}\|\boldsymbol{u} - \boldsymbol{w}\| \tag{2.15}$$

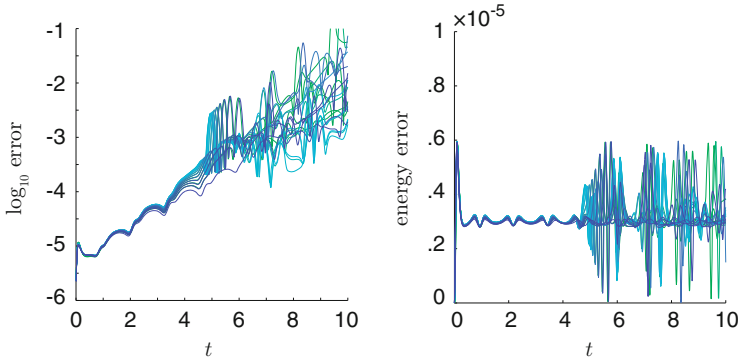

Fig. 2.4 The performance of the Verlet method is demonstrated by repeating the tests of error growth discussed for Euler's method (compare Fig. 2.3). *Left panel*: In the case of the Verlet method, the error is much smaller in an absolute sense than that for the Euler method with the same step-size, but the growth still appears to be exponential. *Right*: the energy error growth as a function of time, very remarkably, compared to the equivalent figure for Euler's method the energy error appears not to grow at all beyond some reasonably small limiting value, suggesting that the Verlet numerical approximation will remain near the desired energy surface

for all $\boldsymbol{u}, \boldsymbol{w}$. Generally speaking this could be taken to hold in a neighborhood of the solution where all approximate solutions for $h < \bar{h}$ are assumed to lie. With a bit of effort, it is then possible to demonstrate the stability condition for the numerical method (see Exercise 4).

In the left-hand panel of Fig. 2.4 we report the Verlet error growth as a function of time for the Lennard-Jones trimer. Verlet is substantially more accurate in this simulation, although the error still appears to grow exponentially.

Due to the chaotic nature of molecular dynamics, which implies a sensitivity to perturbations of the initial condition or the differential equations themselves, it is to be expected that the global error due to using a numerical method will always grow rapidly (exponentially) in time. As we shall see in later chapters, this does not necessarily mean that a long trajectory is entirely without value. In molecular dynamics it turns out that the real importance of the trajectory is that it provides a mechanism for calculating averages that maintain physical parameters. The simplest example of such a parameter is the energy.

What is more relevant for using Verlet to simulate molecular dynamics is the remarkable stability of energy shown in the right-hand panel of Fig. 2.4. Notice that the energy error does not appear to exhibit a steady accumulation in time (unlike for Euler's method, where it exhibited a linear-in-time growth). The explanation for this unexpected behavior lies in the structural properties of the method, a topic we explore in this and the next chapter.

As another illustration of the performance of the Verlet method, we mention that the dynamical trajectories given in the previous chapter (in particular those given Examples 1.8 and 1.9) were computed using this method (with stepsize $h = 0.00001$). In the case of the calculation of the exponential separation of

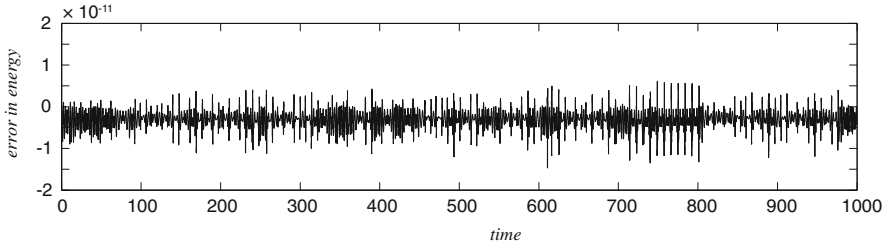

Fig. 2.5 The error in energy in a numerically computed trajectory (Verlet method) with stepsize $h = 0.00001$, for the anisotropic oscillator, showing that it remains bounded by 1.5×10^{-11} in very long runs

trajectories for the anisotropic oscillator shown in Fig. 1.25, the energy errors (see Fig. 2.5) remain bounded and small, suggesting that we may have some confidence in the numerical results presented. This example also provides additional evidence for the lack of "secular growth"[6] of the energy error in Verlet simulations even in a relatively large number (10^8) of timesteps.

2.2.4 *First Integrals and Their Preservation Under Discretization*

Recall that the condition for a given function $I : \mathbb{R}^m \to \mathbb{R}$ to be a first integral is that

$$\nabla I(z) \cdot f(z) = \sum_{j=1}^{m} \frac{\partial I}{\partial z_j}(z) f_j(z) \equiv 0.$$

It is important in this definition that this is an equivalence that holds *everywhere* (or at least in some open set in $\mathbb{R}^m$), so the statement is not just that $I(z(t)) = I(z(0))$ for some particular trajectory, but, moreover, I is conserved for all nearby initial conditions. The flow map preserves the first integral, thus $I(\mathscr{F}_t(z)) \equiv I(z)$. For example, in a Hamiltonian system, the flow map conserves the energy:

$$H \circ \mathscr{F}_t = H.$$

[6]The term "secular growth" in this context is a reference to the long-term growth of perturbations in celestial mechanics. For example, the precession of the Earth's polar axis occurs on a long period relative to its orbital motion and much longer period than its rotation, and so may be classed as a secular motion. In the context of molecular simulations, we use this to refer to accumulation of drift that takes the system steadily away from the energy surface.

There may, in specific instances, be additional first integrals present. For example, in a planar 2-body system in central forces, the angular momentum $xp_y - yp_x = l$ is a conserved quantity, thus $l \circ \mathscr{F}_t = l$.

Now if $I : \mathbb{R}^m \to \mathbb{R}$ has continuous partial derivatives, we may conclude directly that

$$I(\boldsymbol{a}) = I(\boldsymbol{b}) + \nabla I(z_*) \cdot (\boldsymbol{a} - \boldsymbol{b}),$$

where z_* is a point on the line in $\mathbb{R}^m$ connecting $\boldsymbol{a}$ to $\boldsymbol{b}$. Hence

$$|I(\boldsymbol{a}) - I(\boldsymbol{b})| \leq \|\nabla I(z_*)\| \|\boldsymbol{a} - \boldsymbol{b}\|.$$

This means if we know that $\|\nabla I(z)\|$ remains bounded, say less than B, in an open domain containing the solution, we could conclude that the error in I-values computed along a numerical trajectory is of order h^p,

$$|I(z(t_n)) - I(z_n)| \leq \frac{\bar{K}B}{2L} e^{nLh} h^p, \qquad h < \bar{h}, \tag{2.16}$$

with the same assumptions as are needed to characterize the convergence of the method. This also means that we can see a ready means of improving the error in a first integral: simply reduce the timestep. If we have identified a time interval of interest, $[0, \tau]$, then we will automatically have $nh < \tau$ and we need only to choose a stepsize h which brings the error within a target tolerance.

In many cases we will be satisfied with this level of control of the error in the first integrals, but there are a number of limitations of this approach. First, we may find that the constants involved, $\bar{K}$ and, more often, L, are simply so large that it is impossible to conclude anything useful from the bound given above. The stepsize would need to be so preposterously small to make the bound of practical interest that it is better to ignore the bound entirely.

Another limitation may arise when the time interval (the maximum of $\tau = vh$) is very long, as when we use trajectories to sample an energy landscape (the most common use of molecular dynamics). As this enters in the exponent in (2.16) it means that errors may continue to accumulate as we collect more samples and we would eventually leave the acceptable level of deviation in the first integral.

2.3 Geometric Integrators for Hamiltonian Systems

The discussion of the previous section suggests the need for methods with reliable conservation properties. We shall develop methods which, although their errors grow exponentially in time, nonetheless provide excellent energy conservation for very long times. These methods obtain their energy preservation properties indirectly: we design the methods to exactly conserve a certain geometric property

(*symplecticness*) of the phase flow. We will later show (in Chap. 3) that such symplectic methods preserve a perturbed energy-like invariant (to very high accuracy); the preservation of this modified energy then ensures that the original energy is nearly conserved.

The methods considered in this chapter are for the most part very simple in terms of their composition and are based on principles of classical mechanics. The reader may be aware of methods for using variable stepsize, extrapolation, etc. For reasons discussed in the next chapter, these methods and the concerns that motivated their development, which are undoubtedly essential in many applications of ordinary differential equations, are less relevant in the setting of molecular dynamics. The basic issue is that the dynamics of a Hamiltonian system are highly restricted by conservation laws or volume preservation; violation of these principles by naive schemes (which normally do not take advantage of underlying structure) leads to a gradual corruption of the solution. In particular, the errors may accumulate in such a way that the error in energy is severe. It should be stressed that these issues of instability due to violation of structural features of Hamiltonian systems are not always important in small models or for short time computations (where we can just use very small timesteps); historically, they only became apparent as larger, longer simulations began to be performed, as the appetite for numerical data grew.

The framework of geometric integration builds on an understanding of the properties of Hamiltonian mechanics which are well explained in the book of Arnold [15] or in the monograph of Landau and Lifshitz [212].

2.3.1 Volume Preserving Flows: Liouville's Theorem

Consider a set of points $\mathcal{S}(t)$ in phase space with evolution associated to a differential equation $\dot{\boldsymbol{z}} = \boldsymbol{f}(\boldsymbol{z})$ described by the flow map $\mathscr{F}_t(\mathcal{S}(0)) = \mathcal{S}(t)$. Liouville's theorem [16] states that the volume of such a set is invariant with respect to t if the divergence of $\boldsymbol{f}$ vanishes, i.e.

$$\nabla \cdot \boldsymbol{f} = \sum_{i=1}^{m} \frac{\partial f_i}{\partial z_i} = 0.$$

It is a simple exercise to show that for a Hamiltonian system the divergence vanishes, since

$$\nabla \cdot \boldsymbol{f} = \sum_{i=1}^{\mathrm{N_c}} \frac{\partial^2 H}{\partial q_i \partial p_i} - \sum_{i=1}^{\mathrm{N_c}} \frac{\partial^2 H}{\partial p_i \partial q_i} = 0,$$

by equality of mixed partials. Thus Hamiltonian systems always have volume preserving flows.

If we view the map $\mathscr{F}_t$ as a change of variables, we have

$$\mathrm{Vol}(\mathcal{S}(t)) = \int_{\mathcal{S}} |D| \mathrm{d}\omega,$$

where $D = \det\left(\frac{\partial \mathscr{F}_t}{\partial z}\right)$.

To understand where Liouville's theorem comes from, recall that the variational equations of the last chapter are a system of ordinary differential equations for $\boldsymbol{W}(t) = \mathscr{F}_t'(z(t))$:

$$\frac{\mathrm{d}\boldsymbol{W}}{\mathrm{d}t} = \boldsymbol{f}'(z(t))\boldsymbol{W}.$$

Thus

$$\dot{\boldsymbol{W}}\boldsymbol{W}^{-1} = \boldsymbol{f}'(z(t)).$$

Now let $D = \det(\boldsymbol{W})$. One can show (see Exercise 5) that

$$\frac{\dot{D}}{D} = \mathrm{tr}\left(\dot{\boldsymbol{W}}\boldsymbol{W}^{-1}\right).$$

This implies that

$$\dot{D} = \mathrm{div}(\boldsymbol{f}(z(t)))D,$$

and thus

$$D(t) = D(0)e^{\int_0^t \mathrm{div}(\boldsymbol{f}(z(s)))\mathrm{d}s}.$$

In particular, if $\mathrm{div}\boldsymbol{f} \equiv 0$, we see that $D \equiv D(0) = 1$ and it follows that the volume is constant.

Liouville's theorem may be summarized compactly as:

$$\nabla \cdot \boldsymbol{f} = 0 \Rightarrow \det \mathscr{F}_t' = 1.$$

Example 2.3 A 1-d oscillator with Lennard-Jones potential is described by the equations

$$\begin{aligned} \dot{q} &= p, \\ \dot{p} &= -\varphi_{\mathrm{LJ}}'(q). \end{aligned}$$

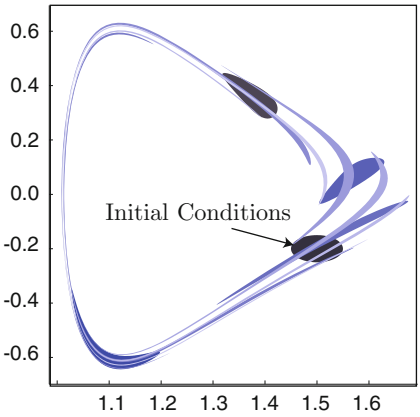

Fig. 2.6 Filamentation of an evolving disk under the flow map of the Lennard-Jones oscillator. Lighter regions represent later images. Despite the increasing complexity of the shapes, each snapshot has the same area

As a consequence of energy conservation, any bounded individual trajectory of this system will be a periodic orbit. Consider the propagation of a small disk of initial conditions,

$$\mathcal{R} = \{(q_0, p_0) | \|(q_0, p_0) - (1.5, -0.2)\| \leq 0.5\}.$$

In Fig. 2.6, we have plotted the sets $\mathscr{F}_t(\mathcal{R})$, for $t = 0, 1, 2, \ldots, 10$, using lighter tones to represent later snapshots. The sets become extremely elongated along the orbital direction.

At all times the energies stay bounded within the interval

$$-0.0960 \leq H(q, p) \leq -0.0669,$$

which is defined by the range of energies in the initial conditions.

Even as the sets become more and more elongated, they always maintain the same total area. This is the direct consequence of Liouville's theorem.

2.3.2 *Volume Preserving Numerical Methods*

The next question is what happens to the volume of a set of points in phase space, which would be conserved by the dynamical system, when we use a numerical method to approximate its evolution.

Consider a linear differential equation system in $\mathbb{R}^m$,

$$\dot{z} = \boldsymbol{S}z,$$

for some matrix $\boldsymbol{S} \in \mathbb{R}^{m \times m}$. The condition for the flow of this system to conserve volume is just that the trace of $\boldsymbol{S}$ (which is the divergence of the vector field

$\boldsymbol{f}(z) = \boldsymbol{S}z$) be zero. Applying Euler's method to the same system results in

$$z_{n+1} = z_n + h\boldsymbol{S}z_n = (\boldsymbol{I} + h\boldsymbol{S})z_n,$$

and the condition for Euler's method to conserve volume is that $\det(\boldsymbol{I} + h\boldsymbol{S}) = 1$. The conditions for volume preservation by the flow map and its Euler approximation are essentially unrelated. Thus Euler's method does not in general conserve phase space volume (it conserves volume only in very special cases—see Exercise 6).

Yet it turns out that there are some numerical methods that always conserve the volume when this is conserved by the flow of the differential equations, or for some particular classes of differential equations. For example, consider the planar system

$$\begin{aligned} \dot{u} &= f(u, v), \\ \dot{v} &= g(u, v), \end{aligned}$$

satisfying the condition $\frac{\partial f}{\partial u} + \frac{\partial g}{\partial v} = 0$, and the asymmetrical variant of Euler's method defined by

$$\begin{aligned} u_{n+1} &= u_n + hf(u_{n+1}, v_n), \\ v_{n+1} &= v_n + hg(u_{n+1}, v_n). \end{aligned}$$

The components of the Jacobian matrix can be obtained using implicit differentiation, they are:

$$a_{11} = \frac{\partial u_{n+1}}{\partial u_n}, \quad a_{12} = \frac{\partial u_{n+1}}{\partial v_n}, \quad a_{21} = \frac{\partial v_{n+1}}{\partial u_n}, \quad a_{22} = \frac{\partial v_{n+1}}{\partial v_n}.$$

So

$$a_{11} = 1 + hf_u a_{11}, \quad a_{12} = hf_u a_{12} + hf_v, \quad a_{21} = hg_u a_{11}, \quad a_{22} = 1 + hg_u a_{12} + hg_v.$$

Solving for the various entries we have

$$\mathscr{G}_h' = \begin{bmatrix} 1/(1 - hf_u) & hf_v/(1 - hf_u) \\ hg_u/(1 - hf_u) & 1 + hg_v + h^2 g_u f_v/(1 - hf_u) \end{bmatrix},$$

and calculating the determinant of the Jacobian results in

$$\det \mathscr{G}_h' = \frac{1 + hg_v}{1 - hf_u}.$$

In the event that the vector field is divergence free, we have $f_u + g_v = 0$ which implies that the numerator and denominator are identical, and it follows that

$\det \mathscr{G}_h' = 1$. Thus the asymmetric variant of the Euler method is area preserving, even though the standard Euler method is not.

It would be valuable to have a general method for deriving volume conserving methods. It turns out that volume conservation is, itself, most readily obtained as a consequence of a more fundamental property of Hamiltonian flows, the conservation of the symplectic form.

2.3.3 The Symplectic Property

Let $m = 2\mathrm{N}_c$. A symplectic map $\boldsymbol{\Phi} : \mathbb{R}^m \to \mathbb{R}^m$ is one that preserves the symplectic differential 2-form. The simplest way to write this is as the following algebraic condition on the Jacobian matrix of $\boldsymbol{\Phi}$:

$$\boldsymbol{\Phi}'^T \boldsymbol{J} \boldsymbol{\Phi}' = \boldsymbol{J}.$$

Recall that the Jacobian matrix is the $m \times m$ matrix of partial derivatives of the components of $\boldsymbol{\Phi}$, $(\boldsymbol{\Phi}')_{ij} = \frac{\partial \Phi_i}{\partial z_j}$, while

$$\boldsymbol{J} = \begin{bmatrix} \mathbf{0} & \boldsymbol{I} \\ -\boldsymbol{I} & \mathbf{0} \end{bmatrix}.$$

So the equation $\boldsymbol{\Phi}'^T \boldsymbol{J} \boldsymbol{\Phi}' = \boldsymbol{J}$ represents a large number ($4\mathrm{N}_c^2$) of equations.

To explain the origin of this formula, we need a brief diversion into the world of differential forms. This language will also be useful in Chap. 4, where we consider constrained systems.

A *1-form* α defined on $\mathbb{R}^m$ is a family of linear mappings from $\mathbb{R}^m$ to $\mathbb{R}$, defined for each point of $\mathbb{R}^m$. Let $\boldsymbol{a} : \mathbb{R}^m \to \mathbb{R}^m$, then we may define a one-form associated to this vector by $\alpha(\boldsymbol{x})(\boldsymbol{\xi}) = \boldsymbol{a}(\boldsymbol{x})^T \boldsymbol{\xi}$.

The *differential* of a function $g : \mathbb{R}^m \to \mathbb{R}$, denoted $\mathrm{d}g$, is a family of linear mappings (one for each point in phase space) from vectors $\boldsymbol{\xi} \in \mathbb{R}^m$ into the reals defined by

$$\mathrm{d}g(\boldsymbol{q}, \boldsymbol{p})(\boldsymbol{\xi}) = \nabla g(\boldsymbol{q}, \boldsymbol{p})^T \boldsymbol{\xi}.$$

So, denoting the ith position coordinate by q_i, we have $\mathrm{d}q_i(\boldsymbol{\xi}) = \xi_i$; the differential is thus an example of a 1-form.

On the other hand the so called *wedge product* of 1-forms α, β, is an example of a *2-form*, at any point in phase space it can be viewed as a quadratic form, i.e. a scalar valued function of two vectors which is linear in each argument. It is written $\alpha \wedge \beta$ and is defined, for vectors $\boldsymbol{\xi}, \boldsymbol{\eta} \in \mathbb{R}^m$ by

$$\alpha \wedge \beta(\boldsymbol{\xi}, \boldsymbol{\eta}) = \alpha(\boldsymbol{\xi})\beta(\boldsymbol{\eta}) - \alpha(\boldsymbol{\eta})\beta(\boldsymbol{\xi}).$$

The wedge product of the coordinate differentials $\mathrm{d}q_i, \mathrm{d}p_i$ may be written

$$\mathrm{d}q_i \wedge \mathrm{d}p_i(\boldsymbol{\xi}, \boldsymbol{\eta}) = \xi_i \eta_{i+\mathrm{N_c}} - \xi_{i+\mathrm{N_c}} \eta_i = \boldsymbol{\xi}^T \boldsymbol{J}^{(i)} \boldsymbol{\eta},$$

where $\boldsymbol{J}^{(i)}$ is the matrix which has zeros everywhere except for $(\boldsymbol{J}^{(i)})_{i,i+\mathrm{N_c}} = 1$, $(\boldsymbol{J}^{(i)})_{i+\mathrm{N_c},i} = -1$. Summing these terms results in the *symplectic 2-form*, denoted ψ_{S}:

$$\psi_{\mathrm{S}} = \sum_{i=1}^{\mathrm{N_c}} \mathrm{d}q_i \wedge \mathrm{d}p_i\,(\boldsymbol{\xi}, \boldsymbol{\eta}) = \boldsymbol{\xi}^T \left(\sum_{i=1}^{\mathrm{N_c}} \boldsymbol{J}^{(i)} \right) \boldsymbol{\eta} = \boldsymbol{\xi}^T \boldsymbol{J} \boldsymbol{\eta}.$$

Thus the matrix

$$\boldsymbol{J} = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{I} \\ -\boldsymbol{I} & \boldsymbol{0} \end{bmatrix}$$

defines a representation for the 2-form ψ_{S}. It is easy to verify that

$$\psi_{\mathrm{S}} = \sum_{i<j} \mathrm{d}z_i \wedge J_{ij} \mathrm{d}z_j.$$

In general, a differential 2-form ψ is represented in coordinates by

$$\psi_z = \sum_{i,j} a_{ij}(z) \mathrm{d}z_i \wedge \mathrm{d}z_j,$$

with matrix of coefficients $\boldsymbol{A}(z) = (a_{ij}(z))$.

The *pull-back* of a differential 1-form ψ under a phase space mapping $\boldsymbol{\Phi}$ is defined as the action of ψ after transformation by the Jacobian matrix of the mapping. It is written $\boldsymbol{\Phi}^* \psi$, so

$$(\boldsymbol{\Phi}^* \psi_z)(\boldsymbol{\xi}) = \psi_{\boldsymbol{\Phi}(z)}(\boldsymbol{\Phi}'(z) \boldsymbol{\xi}).$$

The pull-back of a differential 2-form $\psi_1 \wedge \psi_2$ is consequently defined as

$$\boldsymbol{\Phi}^*(\psi_1 \wedge \psi_2) = (\boldsymbol{\Phi}^* \psi_1) \wedge (\boldsymbol{\Phi}^* \psi_2).$$

Given a differential 2-form ψ_z represented by the matrix $\boldsymbol{A}(z) = (a_{ij}(z))$, the *pull-back* of ψ_z under $\boldsymbol{\Phi}$ is defined by

$$\boldsymbol{\Phi}^* \psi_z = \sum_{ij} b_{ij}(z) \mathrm{d}z_i \wedge \mathrm{d}z_j,$$

where the matrix $\boldsymbol{B}(z) = (b_{ij}(z))$ is related to $\boldsymbol{A}(z)$ by

$$\boldsymbol{B}(z) = \boldsymbol{\Phi}'^{T}(z)\boldsymbol{A}(\boldsymbol{\Phi}(z))\boldsymbol{\Phi}'(z).$$

We say that a 2-form ψ is *conserved under mapping* $\boldsymbol{\Phi}$ if

$$\boldsymbol{\Phi}^{*}\psi = \psi.$$

In coordinates, the conservation of the 2-form represented by matrix $\boldsymbol{A}$ under a mapping $\boldsymbol{\Phi}$ means that

$$\boldsymbol{\Phi}'^{T}\boldsymbol{A}\boldsymbol{\Phi}' = \boldsymbol{A}.$$

In the particular case of the symplectic 2-form ψ_{S}, we have $\boldsymbol{A} = \boldsymbol{J}$, and the following condition for conservation under the mapping $\boldsymbol{\Phi}$

$$\boldsymbol{\Phi}'^{T}\boldsymbol{J}\boldsymbol{\Phi}' = \boldsymbol{J}. \tag{2.17}$$

A map that conserves the symplectic 2-form, or, in coordinates, satisfies (2.17), is termed a *symplectic map*.

Taking the determinant of both sides of (2.17), we have

$$\det(\boldsymbol{\Phi}'^{T}\boldsymbol{J}\boldsymbol{\Phi}') = \det(\boldsymbol{J}) \Rightarrow \det(\boldsymbol{\Phi}'^{T})\det(\boldsymbol{J})\det(\boldsymbol{\Phi}') = \det(\boldsymbol{J}),$$

hence

$$\det(\boldsymbol{\Phi}')^{2} = 1,$$

so $|\det(\boldsymbol{\Phi}')| = 1$. This raises a curious issue since we could conceivably have $\det(\boldsymbol{\Phi}') = -1$.

In the case of a flow map $\boldsymbol{\Phi} = \mathscr{F}_t$, we know that for $t \to 0$, the map reduces to the identity map (the same has to happen for a consistent numerical method). This means that we would have to have

$$\lim_{t\to 0} \det(\mathscr{F}_t') = 1.$$

If the system is Hamiltonian, the map is symplectic for all t, and the determinant will be a continuous function of t, so the cases of interest have $\det(\mathscr{F}_t') = +1$. The flow map of a Hamiltonian system is volume preserving.

2.3.4 Hamiltonian Flow Maps are Symplectic

As we saw at the end of the last chapter, the variational equations describe the change in an infinitesimal perturbation of a solution of a dynamical system. For the Hamiltonian system $\dot{z} = \boldsymbol{J}\nabla H(z)$, these take the form:

$$\dot{\boldsymbol{W}} = \boldsymbol{J}\boldsymbol{S}(t)\boldsymbol{W},$$

where $\boldsymbol{S}(t) = H_{zz}(\boldsymbol{z}(t, \boldsymbol{\zeta}))$ is a symmetric matrix. Computing

$$\boldsymbol{W}^T\boldsymbol{J}\dot{\boldsymbol{W}} = \boldsymbol{W}^T\boldsymbol{J}^2\boldsymbol{S}\boldsymbol{W} = -\boldsymbol{W}^T\boldsymbol{S}\boldsymbol{W},$$

whereas

$$\dot{\boldsymbol{W}}^T\boldsymbol{J}\boldsymbol{W} = \boldsymbol{W}^T\boldsymbol{S}^T\boldsymbol{J}^T\boldsymbol{J}\boldsymbol{W} = \boldsymbol{W}^T\boldsymbol{S}\boldsymbol{W},$$

hence

$$\frac{d}{dt}\boldsymbol{W}^T\boldsymbol{J}\boldsymbol{W} = \boldsymbol{W}^T\boldsymbol{J}\dot{\boldsymbol{W}} + \dot{\boldsymbol{W}}^T\boldsymbol{J}\boldsymbol{W} = 0.$$

This means that $\boldsymbol{W}^T\boldsymbol{J}\boldsymbol{W}$ is a constant matrix. Observe that $\boldsymbol{W}(t) = \mathscr{F}_t'(\boldsymbol{z}(t, \boldsymbol{\zeta}))$ and that $\boldsymbol{W}(0) = \mathscr{F}_t'(\boldsymbol{z}(t, \boldsymbol{\zeta}))\big|_{t=0} = I$, hence

$$\boldsymbol{W}^T\boldsymbol{J}\boldsymbol{W} \equiv \boldsymbol{W}(0)^T\boldsymbol{J}\boldsymbol{W}(0) = \boldsymbol{J}.$$

This proves that the flow map of a Hamiltonian system is a symplectic map.

2.3.5 The Symplectic Maps Form a Group

Let $\boldsymbol{\Phi}_1$ and $\boldsymbol{\Phi}_2$ be any pair of symplectic maps. Then

$$(\boldsymbol{\Phi}_1 \circ \boldsymbol{\Phi}_2)' = \boldsymbol{\Phi}_1'\boldsymbol{\Phi}_2',$$

by the chain rule, hence

$$[(\boldsymbol{\Phi}_1 \circ \boldsymbol{\Phi}_2)']^T\boldsymbol{J}(\boldsymbol{\Phi}_1 \circ \boldsymbol{\Phi}_2)' = [\boldsymbol{\Phi}_1'\boldsymbol{\Phi}_2']^T\boldsymbol{J}\boldsymbol{\Phi}_1'\boldsymbol{\Phi}_2' = \boldsymbol{\Phi}_2'^T\boldsymbol{\Phi}_1'^T\boldsymbol{J}\boldsymbol{\Phi}_1'\boldsymbol{\Phi}_2' = \boldsymbol{\Phi}_2'^T\boldsymbol{J}\boldsymbol{\Phi}_2' = \boldsymbol{J}.$$

Thus the composition of any pair of symplectic maps is a symplectic map. The determinant of a symplectic map is ± 1, hence these maps are always invertible, and the inverse of a symplectic map is symplectic since $\boldsymbol{\Phi}'^T\boldsymbol{J}\boldsymbol{\Phi}' = \boldsymbol{J}$ implies $\boldsymbol{J} = \boldsymbol{\Phi}'^{-T}\boldsymbol{J}\boldsymbol{\Phi}'^{-1}$. Thus the symplectic maps form a group under composition.

2.3.6 Symplectic Integrators

A symplectic integrator is an approximation of the flow map that conserves the symplectic 2-form. Some first steps in the theory of symplectic integration, i.e. numerical methods explicitly designed to mimic the symplectic property of the flow map, were made by Vogalaere [98] (in 1956!) but this work went unnoticed. The first practical methods conserving the symplectic property were suggested by Ruth in 1983 [320] and followed by a number of works on a similar theme [71, 72, 129–131, 267, 353]. Later works, e.g. [132, 139, 214, 325, 398] were aimed at developing methods with higher order of accuracy or better understanding of the meaning of the symplectic property (we will discuss this aspect in the next chapter). Some symplectic integrators are found within existing families (like Runge-Kutta methods), but the most useful are typically obtained using a splitting and composition framework that allows us to build families of such methods.

In the sequel we will write $\boldsymbol{Z} = \mathscr{G}_h(\boldsymbol{z})$ to specify the starting point $\boldsymbol{z}$ and ending point $\boldsymbol{Z}$ of a step.

Alternatively, if we wish to emphasize the decomposition into positions and momenta, we write

$$\begin{bmatrix} \boldsymbol{Q} \\ \boldsymbol{P} \end{bmatrix} = \mathscr{G}_h\left(\begin{bmatrix} \boldsymbol{q} \\ \boldsymbol{p} \end{bmatrix}\right).$$

Recall that, in terms of $\boldsymbol{q}$ and $\boldsymbol{p}$, the differential equations take the form

$$\dot{\boldsymbol{q}} = \nabla_p H, \qquad \dot{\boldsymbol{p}} = -\nabla_q H.$$

In molecular dynamics, the Hamiltonian is usually of the form

$$H = \boldsymbol{p}^T \boldsymbol{M}^{-1} \boldsymbol{p}/2 + U(\boldsymbol{q}),$$

with $\boldsymbol{M}$ a diagonal mass matrix, and we will concentrate on this case for the moment. In this case,

$$\dot{\boldsymbol{q}} = \boldsymbol{M}^{-1}\boldsymbol{p}, \qquad \dot{\boldsymbol{p}} = -\nabla_q U(\boldsymbol{q}) \equiv \boldsymbol{F}(\boldsymbol{q}).$$

The following scheme is a slight modification of the Euler method.

$$\boldsymbol{Q} = \boldsymbol{q} + h\boldsymbol{M}^{-1}\boldsymbol{P}, \tag{2.18}$$

$$\boldsymbol{P} = \boldsymbol{p} + h\boldsymbol{F}(\boldsymbol{q}). \tag{2.19}$$

The method is explicit: to advance the timestep, we use the second equation to compute $\boldsymbol{P}$ and then insert this in the first to get $\boldsymbol{q}$. Let the vectors $\boldsymbol{q}, \boldsymbol{p}, \boldsymbol{Q}, \boldsymbol{P}$ have ith

components q_i, p_i, Q_i, P_i, respectively; we may think of Q_i and P_i as functions of $\boldsymbol{q}$ and $\boldsymbol{p}$, then

$$\mathrm{d}Q_i = \mathrm{d}q_i + hm_i^{-1}\mathrm{d}P_i, \tag{2.20}$$

$$\mathrm{d}P_i = \mathrm{d}p_i - h\sum_{j=1}^{\mathrm{N_c}} \frac{\partial^2 U}{\partial q_j \partial q_i}\mathrm{d}q_j. \tag{2.21}$$

Computing the 2-form,

$$\mathrm{d}Q_i \wedge \mathrm{d}P_i = \mathrm{d}q_i \wedge \mathrm{d}P_i + hm_i^{-1}\mathrm{d}P_i \wedge \mathrm{d}P_i,$$

but $\mathrm{d}u \wedge \mathrm{d}u \equiv 0$ for any u, hence

$$\mathrm{d}Q_i \wedge \mathrm{d}P_i = \mathrm{d}q_i \wedge \mathrm{d}P_i,$$

and (2.21) implies

$$\mathrm{d}Q_i \wedge \mathrm{d}P_i = \mathrm{d}q_i \wedge \mathrm{d}p_i - h\sum_{j=1}^{\mathrm{N_c}} \mathrm{d}q_i \wedge \frac{\partial^2 U}{\partial q_j \partial q_i}\mathrm{d}q_j.$$

It is then a simple exercise to show that

$$\sum_{i=1}^{\mathrm{N_c}}\sum_{j=1}^{\mathrm{N_c}} \mathrm{d}q_i \wedge \frac{\partial^2 U}{\partial q_j \partial q_i}\mathrm{d}q_j = 0,$$

using the skew-symmetry of the wedge product and the fact that the Hessian matrix is symmetric. This implies that

$$\sum_{i=1}^{\mathrm{N_c}} \mathrm{d}Q_i \wedge \mathrm{d}P_i = \sum_{i=1}^{\mathrm{N_c}} \mathrm{d}q_i \wedge \mathrm{d}p_i,$$

which means that the method is symplectic.

2.3.7 The Adjoint Method

Given any numerical integrator $\mathscr{G}_h$, consider the map

$$\mathscr{G}_h^{\dagger} = \mathscr{G}_{-h}^{-1}.$$

This is popularly referred to as the *adjoint* of $\mathscr{G}_h$, although it seems something of an abuse of mathematical language to refer to it in this way. For the flow map $\mathscr{F}_h$, we know that the inverse map is precisely $\mathscr{F}_{-h}$, so $\mathscr{F}_h^\dagger = \mathscr{F}_h$, i.e. the flow map is in the normal sense "self-adjoint," i.e. *symmetric*. However, such a property does not hold in the general case. In particular, consider Euler's method

$$\boldsymbol{Z} = z + h\boldsymbol{f}(z).$$

The adjoint method is defined by

$$\boldsymbol{Z} = z + h\boldsymbol{f}(\boldsymbol{Z}),$$

and where the first was explicit, the second is implicit (it is the so-called *backward Euler* method).

The method (2.18)–(2.19) is called the Symplectic Euler method. Its adjoint method has a similar structure:

$$\boldsymbol{Q} = \boldsymbol{q} + h\boldsymbol{M}^{-1}\boldsymbol{p}, \tag{2.22}$$

$$\boldsymbol{P} = \boldsymbol{p} + h\boldsymbol{F}(\boldsymbol{Q}). \tag{2.23}$$

Comparing with (2.18)–(2.19), we see that (2.22)–(2.23) is also explicit.

Given a method $\mathscr{G}_h$ with adjoint method $\mathscr{G}_h^\dagger$, it is possible to obtain the adjoint of the adjoint method $\mathscr{G}_h^{\dagger\dagger}$, but, as we might expect, the adjoint of the adjoint is the original method:

$$\mathscr{G}_h^{\dagger\dagger} = [\mathscr{G}_{-h}^\dagger]^{-1} = [\mathscr{G}_h^{-1}]^{-1} = \mathscr{G}_h.$$

2.4 Building Symplectic Integrators

Symplectic integrators may be constructed in several ways. First, we may look within standard classes of methods such as the family of Runge-Kutta schemes to see if there are choices of coefficients which make the methods automatically conserve the symplectic 2-form. A second, more direct approach is based on *splitting*. The idea of splitting methods, often referred to in the literature as *Lie-Trotter methods*, is that we divide the Hamiltonian into parts, and determine the flow maps (or, in some cases, approximate flow maps) for the parts, then compose the maps to define numerical methods for the whole system.

2.4.1 Splitting Methods

Let $H(\boldsymbol{q},\boldsymbol{p}) = H_1(\boldsymbol{q},\boldsymbol{p}) + H_2(\boldsymbol{q},\boldsymbol{p})$ have flow map $\mathscr{F}_h$, and indicate by $\mathscr{F}_h^1$, $\mathscr{F}_h^2$ the flow maps for the systems with Hamiltonians H_1, H_2, respectively. The proposal is that the map

$$\mathscr{G}_h = \mathscr{F}_h^1 \circ \mathscr{F}_h^2$$

is an approximation of $\mathscr{F}_h$. For this to be a first order method, we need at least

$$\|\mathscr{G}_h(\boldsymbol{u}) - \mathscr{F}_h(\boldsymbol{u})\| \leq C(\boldsymbol{u})h^2.$$

Expand $\mathscr{F}_h(\boldsymbol{u})$ in a Taylor series:

$$\mathscr{F}_h(\boldsymbol{u}) = \boldsymbol{u} + h\boldsymbol{J}\nabla H + \mathcal{O}(h^2) = \boldsymbol{u} + h(\boldsymbol{J}\nabla H_1 + \boldsymbol{J}\nabla H_2) + \mathcal{O}(h^2). \tag{2.24}$$

On the other hand,

$$\mathscr{F}_h^1(\boldsymbol{u}) = \boldsymbol{u} + h\boldsymbol{J}\nabla H_1(\boldsymbol{u}) + \mathcal{O}(h^2), \qquad \mathscr{F}_h^2(\boldsymbol{u}) = \boldsymbol{u} + h\boldsymbol{J}\nabla H_2(\boldsymbol{u}) + \mathcal{O}(h^2).$$

Composing the maps we have

$$\mathscr{F}_h^1 \circ \mathscr{F}_h^2(\boldsymbol{u}) = \boldsymbol{u} + h\boldsymbol{J}\nabla H_2(\boldsymbol{u}) + h\boldsymbol{J}\nabla H_1(\boldsymbol{u} + h\boldsymbol{J}\nabla H_2(\boldsymbol{u})) + \mathcal{O}(h^2).$$

Assuming H_1 is C^2, we have

$$\mathscr{F}_h^1 \circ \mathscr{F}_h^2(\boldsymbol{u}) = \boldsymbol{u} + h(\boldsymbol{J}\nabla H_2(\boldsymbol{u}) + \boldsymbol{J}\nabla H_1(\boldsymbol{u})) + \mathcal{O}(h^2),$$

which agrees with the expansion (2.24) of the flow map through the terms of first order. Thus the splitting method does indeed provide a second order local approximation of the flow map.

Example 2.4 Let

$$H_1(\boldsymbol{q},\boldsymbol{p}) = \boldsymbol{p}^T\boldsymbol{M}^{-1}\boldsymbol{p}/2, \qquad H_2(\boldsymbol{q},\boldsymbol{p}) = U(\boldsymbol{q}),$$

then a splitting method for $H = H_1 + H_2$ may be obtained by determining the flow maps for each of the two parts. For H_1 we have the differential equations

$$\dot{\boldsymbol{q}} = \boldsymbol{M}^{-1}\boldsymbol{p}, \qquad \dot{\boldsymbol{p}} = \boldsymbol{0}.$$

The second equation tells us that p is constant, so it is fixed at its initial value, whereas the first equation says that $\boldsymbol{q}$ evolves on a linear path, hence its flow map is

$$\begin{aligned} \boldsymbol{Q} &= \boldsymbol{q} + h\boldsymbol{M}^{-1}\boldsymbol{p}, \\ \boldsymbol{P} &= \boldsymbol{p}. \end{aligned}$$

Similarly, the flow map of the system with Hamiltonian $H_2 = U$ is

$$\begin{aligned} \boldsymbol{Q} &= \boldsymbol{q}, \\ \boldsymbol{P} &= \boldsymbol{p} - h\nabla U(\boldsymbol{q}). \end{aligned}$$

The composition of these maps simplifies to

$$\begin{aligned} \boldsymbol{Q} &= \boldsymbol{q} + h\boldsymbol{M}^{-1}\boldsymbol{P}, \\ \boldsymbol{P} &= \boldsymbol{p} - h\nabla U(\boldsymbol{q}), \end{aligned}$$

which is precisely the Symplectic Euler method. Similarly, composing the same two maps in the opposite order gives the adjoint Symplectic Euler method.

Example 2.5 (Verlet is Symplectic) For the Symplectic Euler method $\mathscr{G}_h$ and its adjoint method $\mathscr{G}_h^\dagger$, consider the composition

$$\mathcal{K}_h := \mathscr{G}_{h/2}^\dagger \circ \mathscr{G}_{h/2}.$$

Let

$$\begin{bmatrix} \bar{\boldsymbol{q}} \\ \bar{\boldsymbol{p}} \end{bmatrix} = \mathscr{G}_{h/2} \begin{bmatrix} \boldsymbol{q} \\ \boldsymbol{p} \end{bmatrix},$$

i.e.,

$$\bar{\boldsymbol{q}} = \boldsymbol{q} + (h/2)\boldsymbol{M}^{-1}\bar{\boldsymbol{p}}, \qquad \bar{\boldsymbol{p}} = \boldsymbol{p} - (h/2)\nabla U(\boldsymbol{q}).$$

and set

$$\begin{bmatrix} \boldsymbol{Q} \\ \boldsymbol{P} \end{bmatrix} = \mathscr{G}_{h/2}^\dagger \begin{bmatrix} \bar{\boldsymbol{q}} \\ \bar{\boldsymbol{p}} \end{bmatrix},$$

i.e.,

$$\boldsymbol{Q} = \bar{\boldsymbol{q}} + (h/2)\boldsymbol{M}^{-1}\bar{\boldsymbol{p}}, \qquad \boldsymbol{P} = \bar{\boldsymbol{p}} - (h/2)\nabla U(\boldsymbol{Q}).$$

The composition simplifies to

$$\begin{aligned} \bar{\boldsymbol{p}} &= \boldsymbol{p} - \frac{h}{2}\nabla U(\boldsymbol{q}), \\ \boldsymbol{Q} &= \boldsymbol{q} + h\boldsymbol{M}^{-1}\bar{\boldsymbol{p}}, \\ \boldsymbol{P} &= \bar{\boldsymbol{p}} - \frac{h}{2}\nabla U(\boldsymbol{Q}). \end{aligned}$$

This is just the leapfrog/Verlet method in its Hamiltonian form. Since we have obtained the method as the composition of two symplectic maps, and the symplectic maps form a group, we know that this method will also be symplectic.

When we construct a composition of a method with its adjoint method, we see readily that it is symmetric, i.e. self-adjoint, since

$$\mathcal{K}_h^\dagger := [\mathcal{G}_{h/2}^\dagger \circ \mathcal{G}_{h/2}]^\dagger = \mathcal{G}_{h/2}^\dagger \circ \mathcal{G}_{h/2} = \mathcal{K}_h.$$

Self-adjoint, or *symmetric* schemes have unique features; in particular they have even order [164, 227].

As we shall see in the next chapter, it is possible to construct methods of arbitrary order by employing more sophisticated multi-stage compositions of mappings.

2.4.2 General Composition Methods

When using splittings, it is not necessary to solve each Hamiltonian of a splitting using the exact flow. Instead, we may replace the flow maps of any part by an approximation. More generally, if we have any two symplectic numerical methods, say $\mathcal{G}_h^1$ and $\mathcal{G}_h^2$, then the composition

$$\mathcal{G}_h := \mathcal{G}_{h/2}^1 \circ \mathcal{G}_{h/2}^2$$

is another symplectic numerical method. The order of this method is typically the minimum of the orders of the two methods involved, but it can be higher, as the example of the Verlet method (constructed by composing symplectic Euler and its adjoint) shows.

2.4.3 Harmonic + Anharmonic Splitting

Some systems can be decomposed into a harmonic (quadratic) part $H_0(\boldsymbol{q},\boldsymbol{p}) = \boldsymbol{p}^T\boldsymbol{M}^{-1}\boldsymbol{p}/2 + \boldsymbol{q}^T\boldsymbol{A}\boldsymbol{q}/2$ and an anharmonic part $H_1(\boldsymbol{q},\boldsymbol{p}) = \hat{U}(\boldsymbol{q})$. Then a splitting method can be formulated based on exact solution of the linear system together with "kicks" representing impulses defined by the anharmonic part. For a simple system consisting of a harmonic oscillator $H_0(q,p) = p^2/2 + \Omega^2 q^2/2$, such a method can be written

$$\begin{bmatrix} Q \\ \hat{p} \end{bmatrix} = \begin{bmatrix} \cos h\Omega & \frac{1}{\Omega}\sin h\Omega \\ -\Omega \sin h\Omega & \cos h\Omega \end{bmatrix} \begin{bmatrix} q \\ p \end{bmatrix},$$
$$P = \hat{p} - h\hat{U}'(Q).$$

2.4.4 Explicit and Implicit Methods

All of the methods mentioned so far are said to be *explicit* discretizations, since producing the next approximation point on a trajectory does not require solving any implicit equations defined in terms of the previous one. The Verlet method is not implicit even though $\boldsymbol{Q}$ appears on the right in the second equation, since it is given in an explicit way in terms of $\boldsymbol{q}$ and $\boldsymbol{p}$. Implicitness adds another layer in both analysis and numerical implementation, which, however, in certain applications, is readily justified.

Example 2.6 The *Backward Euler* method,

$$z_{n+1} = z_n + h\boldsymbol{f}(z_{n+1}),$$

is an example of an implicit method. The calculation of a timestep involves solving a system of equations of the form

$$\boldsymbol{g}(\boldsymbol{w}) \equiv \boldsymbol{w} - z_n - h\boldsymbol{f}(\boldsymbol{w}) = 0.$$

The map $\mathscr{G}_t$ is defined implicitly by the equation

$$\mathscr{G}_h(z) = z + h\boldsymbol{f}(\mathscr{G}_h(z)).$$

An implicit method will typically result in a system of nonlinear equations of the form

$$\boldsymbol{g}(z_{n+1}) = \boldsymbol{\tau}_n, \tag{2.25}$$

which will need to be solved at each timestep. The right hand side $\boldsymbol{\tau}_n$ is a vector that depends on the previous time-step z_n, perhaps in a complicated way. We may assume the number of equations represented by (2.25) is equal to the dimension of the space where z is defined, so we have a square nonlinear system. Typically $\boldsymbol{g}$ will depend on the stepsize and coefficients of the method and will have the property that for h sufficiently small, the solution is uniquely defined and is continuously defined in terms of $\boldsymbol{\tau}_n$, that is the mapping $\boldsymbol{g}$ has a bounded and smooth inverse.

Solving the system may then proceed, from an initial guess $z_{n+1}^{(0)}$ by use of Newton's method:

$$z_{n+1}^{(k+1)} = z_{n+1}^{(k)} - \left[\boldsymbol{J}^{(k)}\right]^{-1} (\boldsymbol{g}(z_n^{(k)}) - \boldsymbol{\tau}_n),$$

where $\boldsymbol{J}^{(k)}$ is the Jacobian matrix of the mapping $\boldsymbol{g}$ evaluated at $z_n^{(k)}$, or else an approximation of this Jacobian (assumed to be nonsingular due to the invertibility

of the mapping). The iteration may be recast in the form

$$\boldsymbol{b}_k = -(\boldsymbol{g}(z_n^{(k)}) - \boldsymbol{\tau}_n), \qquad \boldsymbol{J}^{(k)} \Delta z_k = \boldsymbol{b}_k, \qquad z_{n+1}^{(k+1)} = z_{n+1}^{(k)} + \Delta z_k.$$

The potentially demanding steps here are the calculation of the Jacobian matrix and the solution of the linear system.

One way to save computational work is to recycle the Jacobian from a previous timestep. Alternatively, it may be possible to approximate the Jacobian crudely in such a way that the iteration still converges. For example, if the Jacobian matrix is large and can be written in the form

$$\boldsymbol{J}(z) = \boldsymbol{D} + \boldsymbol{E}(z),$$

where $\boldsymbol{E}$ is small in norm and $\boldsymbol{D}$ is a constant sparse matrix, then, in many cases, the Jacobian matrix may be replaced by the constant matrix $\boldsymbol{D}$ and the iteration will still converge. Newton's method (without approximation of the Jacobian matrix) has a remarkable *quadratic* convergence property, meaning that, when the initial guess is close the solution, the errors e_k, e_{k+1} at the kth and $k+1$st iterations satisfy the relation

$$e_{k+1} \leq K e_k^2.$$

This rapid convergence is typically lost when the Jacobian matrix is approximated in some way, and one finds instead

$$e_{k+1} \leq \rho e_k,$$

where $0 < \rho < 1$, i.e., quadratic convergence is replaced by geometric convergence.

In some cases *semi-implicit* methods can be developed which may only require the solution of a low-dimensional nonlinear system at each step. In Chap. 4, we discuss constrained systems for which implicit methods are needed and, in the case of the SHAKE method, for which the nonlinear system that must be solved at each step is of dimension equal to the number of constraints imposed. This is an example of a semi-implicit method.

In other cases, the special structure of the underlying problem may lead to certain efficiencies in the implementation, as when a sparse matrix is obtained in the linear equations that must be solved at each timestep.

Other methods may be substituted for Newton's method for the purpose of solving the nonlinear equations [292] of an implicit method.

2.4.5 "Processed" Methods and Conjugacy

One of the intriguing ideas that can be exploited to improve numerical integrators is that of *conjugacy*. In general, we say that two maps $\boldsymbol{A}$ and $\boldsymbol{B}$ are conjugate if there is a homeomorphism[7] $\boldsymbol{\chi}$ such that

$$\boldsymbol{A} = \boldsymbol{\chi}^{-1} \circ \boldsymbol{B} \circ \boldsymbol{\chi}.$$

That is, we may evaluate one map by first transforming the input, then applying the other map, then transforming the output by the inverse of the first transformation. Conjugate maps have the property that their iterates are also conjugate, since

$$\begin{aligned} \boldsymbol{A}^n &= (\boldsymbol{\chi}^{-1} \circ \boldsymbol{B} \circ \boldsymbol{\chi})^n \\ &= (\boldsymbol{\chi}^{-1} \circ \boldsymbol{B} \circ \boldsymbol{\chi}) \circ (\boldsymbol{\chi}^{-1} \circ \boldsymbol{B} \circ \boldsymbol{\chi}) \cdots (\boldsymbol{\chi}^{-1} \circ \boldsymbol{B} \circ \boldsymbol{\chi}) \\ &= \boldsymbol{\chi}^{-1} \circ \boldsymbol{B}^n \circ \boldsymbol{\chi}. \end{aligned}$$

If $\boldsymbol{A}$ and $\boldsymbol{B}$ are maps of phase space, then the conjugacy implies that they have equivalent stability properties under iteration, since if $\boldsymbol{B}^n(z_0) \to z^*$, as $n \to \infty$, for all initial points z_0, then also $\boldsymbol{A}^n(z_0) \to \boldsymbol{\chi}^{-1}(z^*)$.

A common use made of conjugacy in the setting of numerical integration is to increase the order of accuracy. Numerical integrators are dependent on the stepsize h. The idea is to introduce a conjugacy via a map χ_h which also depends on the stepsize and by judicious design of the map, to eliminate leading terms in the expansion of the local error of the method. As an illustration, the Symplectic Euler method turns out to be conjugate to the Verlet method (see Exercise 12). One sometimes refers to the "effective order" of a numerical method as the order attainable via processing, thus the effective order of the Symplectic Euler method would be two.

Let us suppose that we have such a conjugacy between two numerical methods $\mathscr{G}_h$ and $\widetilde{\mathscr{G}}_h$, that is

$$\mathscr{G}_h = \boldsymbol{\chi}_h^{-1} \circ \widetilde{\mathscr{G}}_h \circ \boldsymbol{\chi}_h,$$

defined in such a way that $\mathscr{G}_h$ has order r and $\widetilde{\mathscr{G}}_h$ has order s where $s < r$. Then, given an initial condition z_0, we first modify ("pre-process") this to $\tilde{z}_0 = \boldsymbol{\chi}_h(z_0)$, then take multiple steps with the method $\widetilde{\mathscr{G}}_h$, and finally transform ("post-process") the end result by $\boldsymbol{\chi}_h^{-1}$. The resulting approximation will be of order r even though all timestepping was performed using a lower order method. The situation is diagrammed in Fig. 2.7.

[7] A homeomorphism is a continuous bijection which has a continuous inverse.

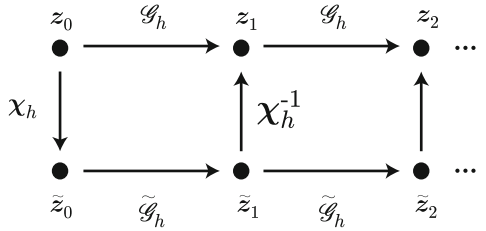

Fig. 2.7 In a "processed method" the inputs are first transformed before repeated application of a given method; after post-processing, the output may have higher order of accuracy than would be obtained using a straightforward application of the scheme

The use of conjugacy as a method to develop enhanced numerical methods was introduced in [397] under the name *symplectic correctors*.

2.5 Other Classes of Methods

2.5.1 Runge-Kutta Methods

The family of Runge-Kutta methods for solving $\dot{z} = \boldsymbol{f}(z)$ is defined by

$$\boldsymbol{Z} = z + h \sum_{i=1}^{s} b_i \boldsymbol{F}_i,$$

where the vectors $\boldsymbol{F}_i$, $i = 1, \ldots, s$, are computed by solving the system

$$\boldsymbol{F}_i = \boldsymbol{f}(z + h \sum_{j=1}^{s} a_{ij} \boldsymbol{F}_j), \qquad i = 1, \ldots, s.$$

The parameters s, $b_1, b_2, \ldots, b_s$, and the $s \times s$ matrix of coefficients $\boldsymbol{A} = (a_{ij})$ describe the method. In some cases the determination of the vectors $\boldsymbol{F}_i$ requires the solution of a nonlinear system, i.e. the method is *implicit*; in other cases the method can be explicit. An example of a popular 4th order explicit method is the choice of matrix $\boldsymbol{A}$ with coefficients $a_{ij} \equiv 0$ except $a_{21} = 1/2$, $a_{32} = 1/2$ and $a_{43} = 1$, and $b_1 = 1/6$, $b_2 = 1/3$, $b_3 = 1/3$, $b_4 = 1/6$. This method is not symplectic, and it is in fact impossible to find symplectic explicit methods within the Runge-Kutta family.

Let us emphasize that, while a typical RK method is not symplectic, **some** implicit Runge-Kutta methods **are** symplectic. The precise condition that must be

satisfied [325] is

$$b_i a_{ij} + b_j a_{ji} = b_i b_j, \qquad i = 1, \ldots, s, \qquad j = 1, \ldots s.$$

The implicitness reduces the usefulness of Runge-Kutta methods for applications like molecular dynamics, because the complexity and cost of the molecular force laws usually means that implicit methods are inefficient (this is certainly not prima facie obvious, but the issue has been examined by several researchers without much positive result; see, e.g. [308]).

Example 2.7 The Gauss-Legendre family of Runge-Kutta (GLRK) methods correspond to approximating the vector field at the Gauss points, i.e. the zeros of the orthogonal polynomials that arise in Gaussian quadrature. As these points are symmetrically distributed the GLRK schemes are symmetric, hence have even order. The simplest such method is the *implicit midpoint* rule:

$$\boldsymbol{Z} = z + h\boldsymbol{F}_1, \qquad \boldsymbol{F}_1 = \boldsymbol{f}(z + \frac{h}{2}\boldsymbol{F}_1),$$

which has order 2. The 4th order method ($s = 2$) has coefficients

$$b_1 = b_2 = \frac{1}{2}, \qquad A = (a_{ij}) = \begin{bmatrix} \frac{1}{4} & \frac{1}{4} - \frac{\sqrt{3}}{6} \\ \frac{1}{4} + \frac{\sqrt{3}}{6} & \frac{1}{4} \end{bmatrix}.$$

2.5.2 Partitioned Runge-Kutta Methods

It can be seen that Runge-Kutta methods treat all components of the differential equation identically. On the other hand, in molecular dynamics the equations of motion often have a special structure: for example the differential equation system is typically linear in $\boldsymbol{p}$, and, moreover, the equations have a special coupling structure so that the differential equation for $\boldsymbol{q}$ depends only on $\boldsymbol{p}$ and that for $\boldsymbol{p}$ depends only on $\boldsymbol{q}$. So-called partitioned Runge-Kutta methods allow us to exploit this structure. As an illustration, consider the method:

$$\hat{\boldsymbol{P}} = \boldsymbol{p} - \frac{h}{2}\nabla_{\boldsymbol{q}} H(\boldsymbol{q}, \hat{\boldsymbol{P}}), \tag{2.26}$$

$$\boldsymbol{Q} = \boldsymbol{q} + \frac{h}{2}(\nabla_{\boldsymbol{p}} H(\boldsymbol{q}, \hat{\boldsymbol{P}}) + \nabla_{\boldsymbol{p}} H(\boldsymbol{Q}, \hat{\boldsymbol{P}})), \tag{2.27}$$

$$\boldsymbol{P} = \hat{\boldsymbol{P}} - \frac{h}{2}\nabla_{\boldsymbol{q}} H(\boldsymbol{Q}, \hat{\boldsymbol{P}}). \tag{2.28}$$

When $H = \boldsymbol{p}^T \boldsymbol{M}^{-1}\boldsymbol{p}/2 + U(\boldsymbol{q})$ this is just the leapfrog/Verlet method, but it can be used also for more general systems. In the more general setting it is implicit (a nonlinear set of equations must be solved to advance from time level to time level). To see that it is symplectic, we first note that this is a symmetric composition of the form

$$\mathcal{K}_h = \mathcal{G}_{h/2}^{\dagger} \circ \mathcal{G}_{h/2},$$

where $\mathcal{G}_h$ is defined by

$$\boldsymbol{P} = \boldsymbol{p} - h\nabla_{\boldsymbol{q}} H(\boldsymbol{q}, \boldsymbol{P}), \tag{2.29}$$

$$\boldsymbol{Q} = \boldsymbol{q} + h\nabla_{\boldsymbol{p}} H(\boldsymbol{q}, \boldsymbol{P}), \tag{2.30}$$

so it is enough to show that this basic method is symplectic. Taking differentials of (2.30) defining $\mathcal{G}_h$ and then wedge products and summing, we have

$$\sum_i \mathrm{d}Q_i \wedge \mathrm{d}P_i = \sum_i \mathrm{d}q_i \wedge \mathrm{d}P_i + h \sum_i \sum_j H_{p_i q_j} \mathrm{d}q_j \wedge \mathrm{d}P_i$$
$$+ h \sum_i \sum_j H_{p_i p_j} \mathrm{d}P_j \wedge \mathrm{d}P_i.$$

The last term on the right vanishes by equality of mixed partials and the antisymmetry of the wedge product. On the other hand, using (2.29), we obtain, by similar means,

$$\sum_i \mathrm{d}q_i \wedge \mathrm{d}P_i = \sum_i \mathrm{d}q_i \wedge \mathrm{d}p_i - h \sum_i \sum_j H_{q_i p_j} \mathrm{d}q_i \wedge \mathrm{d}P_j.$$

Relabelling the indices in this sum and using our previous work results in

$$\sum_i \mathrm{d}Q_i \wedge \mathrm{d}P_i = \sum_i \mathrm{d}q_i \wedge \mathrm{d}p_i,$$

implying that the method is symplectic.

The more general family of Partitioned Runge-Kutta methods is defined by making use of a partitioning of the system and introducing combinations of a set of internal stages. This more general family of schemes is discussed in some detail in [326] (see also discussions of [164, 227]).

2.5.3 Newmark Methods

As a special case of a partitioned Runge-Kutta method, consider the Newmark family of methods [280] defined for two parameters σ and η by the formulas

$$\boldsymbol{P} = \boldsymbol{p} - h(1-\sigma)\nabla U(\boldsymbol{q}) - h\sigma\nabla U(\boldsymbol{Q}),$$
$$\boldsymbol{Q} = \boldsymbol{q} + h\boldsymbol{M}^{-1}\boldsymbol{p} - h^2\left(\frac{1}{2}-\eta\right)\nabla U(\boldsymbol{q}) - h^2\eta\nabla U(\boldsymbol{Q}).$$

This is an exact transcription of the formulas in Newmark's original paper, only substituting $\boldsymbol{M}^{-1}\boldsymbol{p}$ for the velocity $\boldsymbol{v}$ wherever it appears. In practice the choice $\sigma = 1/2$ is used to avoid spurious damping (it can be demonstrated for a simple model problem); this certainly would appear to be desirable in the setting of molecular dynamics. For $\eta = 0$ we then arrive at the Verlet method. For other values of η the scheme is clearly implicit, which likely is the reason it is rarely used in molecular simulation, although it is popular in structural mechanics. The implicit Newmark methods are not symplectic, but a related family of symplectic methods can be constructed by replacing interpolated forces by forces evaluated at interpolated positions [395].

2.5.4 Multiderivative Methods

All of the methods discussed above rely on computing values of the vector field (i.e. momenta and forces) only. However, as we saw earlier in this chapter, in the setting of a Taylor series method, we may approximate a single step by

$$z_{n+1} = z_n + h\dot{z}_n + \frac{h^2}{2}\ddot{z}_n + \ldots + \frac{h^k}{k!}z_n^{(k)},$$

where it is possible to make use of higher order derivatives of the solution in formulating the method. Then using the differential equation, the time derivatives may be replaced by elementary differentials of the vector field. This same idea can be used in a more sophisticated way to improve the accuracy of molecular dynamics methods. Letting $H(\boldsymbol{q},\boldsymbol{p}) = \boldsymbol{p}^T\boldsymbol{M}^{-1}\boldsymbol{p}/2 + U(\boldsymbol{q})$, the Takahashi-Imada method [355] (also known as Rowlands' method [316]) has the same form as the Verlet method

$$\hat{\boldsymbol{P}} = \boldsymbol{p} - (h/2)\nabla\tilde{U}(\boldsymbol{q}),$$
$$\boldsymbol{Q} = \boldsymbol{q} + h\boldsymbol{M}^{-1}\hat{\boldsymbol{P}},$$
$$\boldsymbol{P} = \hat{\boldsymbol{P}} - (h/2)\nabla\tilde{U}(\boldsymbol{Q}),$$

where the corresponding potential energy function is

$$\tilde{U}(\boldsymbol{q}) = U(\boldsymbol{q}) - \frac{h^2}{24}\nabla U(\boldsymbol{q})^T \boldsymbol{M}^{-1}\nabla U(\boldsymbol{q}).$$

The forces arising from such a modified potential can be worked out:

$$\tilde{\boldsymbol{F}} = -\nabla\tilde{U} = -\left[\boldsymbol{I} + \frac{h^2}{12}U''\boldsymbol{M}^{-1}\right]\nabla U,$$

where U'' is the Hessian matrix of the potential. It may in some cases be a daunting task to compute the Hessian matrix as would be required in the Takahashi-Imada method, but this is not always the case. Advantage can be made of the fact that for systems with short-ranged potentials, the Hessian matrix is likely to be very sparse, in which case this computation can be effected with little additional overhead, although this will depend on the underlying computer architecture, the size of the system, etc.

This method can be shown to have effective order four, meaning that there is a change of variables $\boldsymbol{\chi}_h$ which can be used to transform the Takahashi-Imada method into one of order four using the processing technique of Sect. 2.4.5. The potential energy modification has been specifically chosen to annihilate terms in the local error expansion (after coordinate transformation).

A discussion and comparison of several multiderivative methods for molecular applications may be found in [239].

2.5.5 *Other Methods*

We have mentioned previously that it is possible to reduce the Verlet method to a scheme involving positions only:

$$\boldsymbol{q}_{n+1} - 2\boldsymbol{q}_n + \boldsymbol{q}_{n-1} = h^2\boldsymbol{M}^{-1}\boldsymbol{F}(\boldsymbol{q}_n).$$

This is a type of a *multistep* method. Such methods may be studied using a generalization of the techniques used to understand one-step methods [167]. There are a variety of multistep methods which could in principle be used for molecular dynamics, however, we regard the benefits as unproven; in particular, such methods neglect the phase space structure such as the symplectic property.

Another scheme which is sometimes used in molecular dynamics is Beeman's Algorithm [30, 331].

Example 2.8 (Beeman's Algorithm) The method treats the positions and momenta differently, updating these from the formulas

$$\boldsymbol{q}_{n+1} = \boldsymbol{q}_n + h\dot{\boldsymbol{q}}_n + \frac{h^2}{6}[4\ddot{\boldsymbol{q}}_n - \ddot{\boldsymbol{q}}_{n-1}]. \tag{2.31}$$

$$\boldsymbol{p}_{n+1} = \boldsymbol{p}_n + \frac{h}{6}\boldsymbol{M}[2\ddot{\boldsymbol{q}}_{n+1} + 5\ddot{\boldsymbol{q}}_n - \ddot{\boldsymbol{q}}_{n-1}]. \tag{2.32}$$

The shorthand $\dot{\boldsymbol{q}}_n \equiv \boldsymbol{M}^{-1}\boldsymbol{p}_n$, $\ddot{\boldsymbol{q}}_n \equiv \boldsymbol{M}^{-1}\boldsymbol{F}(\boldsymbol{q}_n)$ has been used.

This method requires that the positions (and forces) be known at two successive points h apart in time in order to initialize the iteration. These might be generated by using the Verlet method or some other self-starting scheme. Beeman's algorithm is explicit since, given $\boldsymbol{q}_n$, $\boldsymbol{q}_{n-1}$ and $\boldsymbol{p}_n$, one directly obtains $\boldsymbol{q}_{n+1}$ and then, $\ddot{\boldsymbol{q}}_{n+1}$, and thus $\boldsymbol{p}_{n+1}$, with only one new force evaluation. Because it is a "partitioned multistep method," its analysis is more involved than for the one-step methods, and, in particular its qualitative features are difficult to relate to those of the flow map. The order of accuracy of the scheme above can be shown to be three.

Exercises

1. The Morse potential is often encountered in molecular modelling and takes the form

$$\varphi_{\text{Morse}}(r) = D\left(1 - e^{-a(r-r_e)}\right)^2.$$

Consider a one-particle oscillator in the Morse potential (Hamiltonian $H = p^2/2 + \varphi_{\text{Morse}}(q)$). Explain why the trajectory of a particle will be bound for all time if the initial energy $H(q_0, p_0) < D$. Implement the 4th order Runge-Kutta and Verlet methods and explore the problem for $a = 1, D = 1, r_e = 1$. You should observe that all trajectories of the Runge-Kutta method eventually diverge. Study the dissociation time (the time until the particle is ejected to some fixed large distance, say $r = 3$) as a function of the stepsize and the initial energy.

2. Demonstrate the bound (2.12). [Hint: use $e^{Lh} > 1 + hL$.]

3. Let f satisfy the Lipshitz condition $|f(u) - f(v)| \leq L|u - v|$. Let $\mathscr{G}_h$ represent the flow map approximation associated to Euler's method. Show that

$$|\mathscr{G}_h(u) - \mathscr{G}_h(v)| \leq (1 + hL)|f(u) - f(v)|.$$

Next prove (2.10) holds by finding Δ, $\bar{K}$ and p. This can be used to prove Theorem 2.1.

4. Let $\mathscr{G}_t$ denote the Verlet approximation of the flow map defined by (2.13)–(2.14). Using the Lipschitz condition on $\boldsymbol{F}$ (2.15), show that, for h sufficiently small,

$$\|\mathscr{G}_h(\boldsymbol{u}) - \mathscr{G}_h(\boldsymbol{v})\| \leq (1 + hR)\|\boldsymbol{u} - \boldsymbol{v}\|,$$

and find R.

5. Let $\boldsymbol{W} = \mathscr{F}_t'$ and let $D = \det(\boldsymbol{W})$. We will aim to show that

$$\frac{\dot{D}}{D} = \operatorname{tr}\left(\dot{\boldsymbol{W}}\boldsymbol{W}^{-1}\right), \tag{2.33}$$

which is the key step in the proof of Liouville's theorem. To prove this, recall that the determinant of a matrix can be defined in terms of a co-factor expansion (expansion by minors). If $\boldsymbol{A} = (a_{ij})$ is a given $m \times m$ matrix, then for any $i \in \{1, 2, \ldots, m\}$ we have

$$\det\boldsymbol{A} = \sum_{j=1}^{m} a_{ij}\hat{A}_{ij},$$

where $\hat{A}_{ij}$ is $(-1)^{i+j}$ times the determinant of the $(m-1) \times (m-1)$ dimensional matrix obtained from $\boldsymbol{A}$ by crossing out the ith row and jth column of $\boldsymbol{A}$. Using the cofactor expansion we can easily carry out a proof of the desired relation as follows:

a. Let $\boldsymbol{W} = (w_{ij})$. Show that

$$\frac{\partial \det(\boldsymbol{W})}{\partial w_{ij}} = \hat{W}_{ij},$$

where $\hat{W}_{ij}$ is the ij-co-factor of $\boldsymbol{W}$.

b. Show that

$$\dot{D} = \sum_{i=1}^{m}\sum_{j=1}^{m} \hat{W}_{ij}\dot{w}_{ij}.$$

c. On the other hand, the inverse matrix of $\boldsymbol{W}$ can also be defined in terms of cofactors (the adjugate). If $\boldsymbol{W}^{-1} = (\eta_{ij})$, then

$$\eta_{ij} = \frac{\hat{W}_{ji}}{\det(\boldsymbol{W})}.$$

Demonstrate that this implies (2.33).

6. Euler's method is not symplectic for a general Hamiltonian system. Similarly for a general divergence free vector field, Euler's method is not volume preserving. Find conditions on the vector field that imply that Euler's method is volume preserving. Are there special Hamiltonian systems for which Euler's method is a symplectic method?

7. Let U be a C^2 potential energy function in $\mathrm{N_c}$ variables. Show that

$$\sum_{i=1}^{\mathrm{N_c}} \sum_{j=1}^{\mathrm{N_c}} \mathrm{d}q_i \wedge \frac{\partial^2 U}{\partial q_j \partial q_i} \mathrm{d}q_j = 0,$$

using the skew-symmetry of the wedge product and the fact that the Hessian matrix is symmetric.

8. Is it true that any symplectic method will have a symplectic adjoint method? Either give a proof or find a counterexample.

9. Consider the linear map defined by

$$\boldsymbol{\Phi}(z) = \boldsymbol{R}z, \qquad \boldsymbol{R} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}.$$

Show that, for this to be symplectic, we must have:

$$\begin{bmatrix} a_{11}(a_{21} - a_{12}) & a_{11}a_{22} - a_{12}^2 \\ a_{21}^2 - a_{22}a_{11} & a_{21}a_{22} - a_{22}a_{12} \end{bmatrix} = \boldsymbol{J}.$$

10. Show that the implicit midpoint method

$$z_{n+1} = z_n + h\boldsymbol{f}\left(\frac{z_n + z_{n+1}}{2}\right)$$

is symplectic. Hint: let $\bar{z} = \frac{z_n + z_{n+1}}{2}$ and use the properties of the wedge product to show that

$$\mathrm{d}z_{n+1} \wedge \boldsymbol{J}\mathrm{d}z_{n+1} = \mathrm{d}z_n \wedge \boldsymbol{J}\mathrm{d}z_n.$$

11. Considering a simple model for a bound pair of atoms with positions q_1 and q_2, moving in a position dependent potential field φ, with Hamiltonian

$$H(\boldsymbol{q},\boldsymbol{p}) = \frac{\|\boldsymbol{p}_1\|^2}{2m_1} + \frac{\|\boldsymbol{p}_2\|^2}{2m_2} + \varphi(\boldsymbol{q}_1) + \varphi(\boldsymbol{q}_2) + \frac{\kappa_b}{2}(\|\boldsymbol{q}_1 - \boldsymbol{q}_2\| - l_b)^2,$$

where κ_b and l_b are the bond vibration coefficient and bond length, respectively. Explain how to treat this problem using a splitting method which involves an exact solution of the isolated problem in the absence of the potential field φ.

12. Show that the Symplectic Euler method is conjugate to the Verlet method.

http://www.springer.com/978-3-319-16374-1

Molecular Dynamics
With Deterministic and Stochastic Numerical Methods
Leimkuhler, B.; Matthews, C.
2015, XXII, 443 p. 95 illus., 71 illus. in color., Hardcover
ISBN: 978-3-319-16374-1