Acta Numerica (2018), pp. 113–206
doi:10.1017/S0962492917000101

Geometric integrators and the Hamiltonian Monte Carlo method

Nawaf Bou-Rabee
Department of Mathematical Sciences,
Rutgers University Camden,
311 N. Fifth Street,
Camden, NJ 08102, USA
E-mail: nawaf.bourabee@rutgers.edu

J. M. Sanz-Serna
Departamento de Matemáticas,
Universidad Carlos III de Madrid,
Avenida de la Universidad 30,
E–28911 Leganés (Madrid), Spain
E-mail: jmsanzserna@gmail.com

This paper surveys in detail the relations between numerical integration and the Hamiltonian (or hybrid) Monte Carlo method (HMC). Since the computational cost of HMC mainly lies in the numerical integrations, these should be performed as efficiently as possible. However, HMC requires methods that have the geometric properties of being volume-preserving and reversible, and this limits the number of integrators that may be used. On the other hand, these geometric properties have important quantitative implications for the integration error, which in turn have an impact on the acceptance rate of the proposal. While at present the velocity Verlet algorithm is the method of choice for good reasons, we argue that Verlet can be improved upon. We also discuss in detail the behaviour of HMC as the dimensionality of the target distribution increases.

CONTENTS

1. Introduction

This paper surveys the relations between numerical integration and the Hamiltonian (or hybrid) Monte Carlo method (HMC), an important and widely used Markov chain Monte Carlo algorithm (Diaconis 2009) for sampling from probability distributions. It is written for a general audience and requires no background on numerical algorithms for solving differential equations. We hope that it will be useful to mathematicians, statisticians and scientists, especially because the efficiency of HMC is largely dependent on the performance of the numerical integrator used in the algorithm.

Named one of the top ten algorithms of the twentieth century (Cipra 2000), MCMC originated in statistical mechanics (Allen and Tildesley 1987, Frenkel and Smit 2002, Krauth 2006, Lelièvre, Rousset and Stoltz 2010, Tuckerman 2010, Landau and Binder 2014) and is now a cornerstone in statistics (Geman and Geman 1984, Gelfand and Smith 1990, Geyer 1992, Tierney 1994); in fact Bayesian approaches only became widespread once MCMC made it possible to sample from completely arbitrary distributions. In conjunction with Bayesian methodologies, MCMC has enabled applications of statistical inference to biostatistics (Jensen, Liu, Zhou and Liu 2004), population modelling (Link and Barker 2009), reliability/risk assessment/uncertainty quantification (Sullivan 2015), machine learning (Andrieu, de Freitas, Doucet and Jordan 2003), inverse problems (Stuart 2010), data assimilation (Chen 2003, Evensen 2009), pattern recognition (Webb 2003, Bishop 2006), artificial intelligence (Ghahramani 2015), and probabilistic robots (Thrun, Burgard and Fox 2005). In these applications, MCMC is used to evaluate the expected values necessary for Bayesian statistical inference, in situations where other methods such as numerical quadrature, Laplace approximation and Monte Carlo importance sampling are impractical or inaccurate. Moreover, MCMC is used as a tool to set the invariant distribution of numerical methods for first- and second-order Langevin stochastic

differential equations (Kikuchi, Yoshida, Maekawa and Watanabe 1991, Roberts and Tweedie 1996*b*, Roberts and Tweedie 1996*a*, Bou-Rabee and Vanden-Eijnden 2010, Bou-Rabee and Vanden-Eijnden 2012, Bou-Rabee and Hairer 2013, Bou-Rabee, Donev and Vanden-Eijnden 2014, Bou-Rabee 2014, Fathi 2014, Fathi, Homman and Stoltz 2015) and stochastic partial differential equations (Beskos, Roberts, Stuart and Voss 2008, Bou-Rabee 2017). Even though MCMC algorithms are often straightforward to program, there are numerous user-friendly, general-purpose software packages available to carry out statistical analysis, including BUGS (Lunn, Thomas, Best and Spiegelhalter 2000, Lunn, Spiegelhalter, Thomas and Best 2009, Lunn *et al.* 2012), Stan (Carpenter *et al.* 2016), MCMCpack (Martin, Quinn and Park 2011), MCMCglmm (Hadfield 2010) and PyMC (Patil, Huard and Fonnesbeck 2010).

HMC itself was invented in 1987 (Duane, Kennedy, Pendleton and Roweth 1987) to study lattice models of quantum field theory, and about a decade later popularized in data science (Liu 2008, Neal 2011). A simple introduction to this algorithm may be found in Sanz-Serna (2014). A key feature of HMC is that it offers the possibility of generating proposal moves that, while being far away from the current state of the Markov chain, may be accepted with high probability, thus avoiding random walk behaviour and reducing the correlation between samples. Such a possibility exists because proposals are generated by numerically integrating a system of Hamiltonian differential equations. The distance between the proposal and the current state may in principle be large if the differential equations are integrated over a suitably long time interval; the acceptance probability of the proposals may be made arbitrarily close to 100% by carrying out the integration with sufficient accuracy. Of particular significance for us is the fact that HMC requires that the numerical integration be performed with a *volume-preserving reversible method*.

Since the computational cost of HMC mainly lies in the numerical integrations, it is of much interest to perform these as efficiently as possible. At present, the well-known velocity Verlet algorithm is the method of choice, but, as will be apparent, Verlet may not be the most efficient integrator one could use. What does it take to design a good integrator for HMC? A key point of this paper is that, due to the specificities of the situation, a number of concepts traditionally used to analyse numerical integrators (including the notions of order of consistency/convergence, error constants, and others) are of limited value in our context. On the one hand and as we have already mentioned, HMC requires methods that have the *geometric properties* of being volume-preserving and reversible, and this limits the number of integrators that may be applied. It is fortunate that, in the last twenty-five years, the literature on the numerical solution of differential equations has given much attention to the construction of integrators

with relevant geometric properties, to the point that *geometric integration* (a term introduced in Sanz-Serna 1997) is now a well-established subfield of numerical analysis (Iserles and Quispel 2017). On the other hand, the properties of preservation of volume and reversibility have important quantitative implications on the integration error (Theorem 6.2), which in turn have an impact on the acceptance rate of proposals. As a consequence, it turns out that, for HMC purposes, the order of the integrator is effectively *twice* its nominal order; for instance the Verlet algorithm behaves, within HMC, as a *fourth-order* integrator. In addition, in HMC, integrators are likely to be operated with large values of the step size, with the implication that analyses that are only valid in the limit of vanishing step size may not be very informative. One has rather to turn to studying the performance of the integrator in well-chosen model problems.

Sections 2–5 provide the necessary background on differential equations, numerical methods, geometric integration and Monte Carlo methods respectively. The heart of the paper is in Section 6. Among the topics considered there, we mention the investigation of the impact on the energy error of the properties of preservation of volume and reversibility (Theorem 6.2) and a detailed study of the behaviour of the integrators in the Gaussian model. Also presented in Section 6 is the construction of integrators more efficient than the Verlet algorithm. Sections 7 and 8 consider, in two different scenarios, the behaviour of HMC as the dimensionality of the target distribution increases. Section 7, based on Beskos *et al.* (2013), studies the model problem where the target is a product of many independent, identical copies. The case where the target arises from discretization of an infinite-dimensional distribution is addressed in Section 8; our treatment, while related to the material in Beskos, Pinski, Sanz-Serna and Stuart (2011), has some novel features because we have avoided the functional analytic language employed in that reference. Finally, Section 9 contains supplementary material.

2. Differential equations and their flows

In this section we introduce some notation and review background material on differential equations, with special emphasis on the Hamiltonian and reversible systems at the basis of HMC algorithms. The section ends with a description of the Lie bracket, which appears in the analysis of the integrators to be used later.

2.1. Preliminaries

We are concerned with autonomous systems of differential equations in $\mathbb{R}^D$,

$$\frac{\mathrm{d}}{\mathrm{d}t}x = f(x); \tag{2.1}$$

the function (vector field) f is assumed throughout to be defined in the whole of $\mathbb{R}^D$ and to be sufficiently smooth. An important role is played by the particular case

$$\frac{\mathrm{d}}{\mathrm{d}t}q = M^{-1}p, \quad \frac{\mathrm{d}}{\mathrm{d}t}p = F(q), \tag{2.2}$$

where $x = (q, p) \in \mathbb{R}^D$, $D = 2d$, $q \in \mathbb{R}^d$, $p \in \mathbb{R}^d$ and M is a constant, invertible matrix, so that $f = (M^{-1}p, F(q))$. By eliminating p, (2.2) is seen to be equivalent to

$$M\frac{\mathrm{d}^2}{\mathrm{d}t^2}q = F(q);$$

this is not the most general autonomous system of second-order differential equations in $\mathbb{R}^d$ because the derivative $(\mathrm{d}/\mathrm{d}t)q$ does not appear on the right-hand side. When the forces depend only on the positions, *Newton's second law* for a mechanical system gives rise to differential equations of the form (2.2); then q, $(\mathrm{d}/\mathrm{d}t)q$, p and F are respectively the vectors of coordinates, velocities, momenta and forces, and M is the matrix of masses.

We let φ_t denote the *t-flow* of the system (2.1) under consideration. By definition, for fixed real t, $\varphi_t : \mathbb{R}^D \to \mathbb{R}^D$ is the map that associates with each $\alpha \in \mathbb{R}^D$ the value at time t of the solution of (2.1) that at the initial time 0 takes the initial value α.

Example 2.1. As a very simple but important example, we consider the standard *harmonic oscillator*, the system in $\mathbb{R}^2$ of the special form (2.2) given by

$$\frac{\mathrm{d}q}{\mathrm{d}t} = p, \quad \frac{\mathrm{d}p}{\mathrm{d}t} = -q. \tag{2.3}$$

For future reference, we note that, with matrix notation, the solutions satisfy

$$\begin{bmatrix} q(t) \\ p(t) \end{bmatrix} = M_t \begin{bmatrix} q(0) \\ p(0) \end{bmatrix}, \quad M_t = \begin{bmatrix} \cos t & \sin t \\ -\sin t & \cos t \end{bmatrix}. \tag{2.4}$$

Thus the flow has the expression

$$\varphi_t(\xi,\ \eta) = (\xi\cos t + \eta\sin t,\ -\xi\sin t + \eta\cos t). \tag{2.5}$$

When $\alpha = (\xi, \eta) \in \mathbb{R}^2$ is fixed and t varies, the right-hand side of (2.5) yields the solution that at $t = 0$ takes the initial value (ξ, η). The notation $\varphi_t(\xi,\ \eta)$ emphasizes that, in the flow, it is the parameter t that is seen as fixed, while $(\xi, \eta) \in \mathbb{R}^2$ is regarded as a variable. Geometrically, φ_t is the clockwise rotation of angle t around the origin of the (q, p)-plane.

For a given system (2.1), it is well possible that for some choices of α and t, the vector $\varphi_t(\alpha) \in \mathbb{R}^D$ is not defined; this will happen if t is outside the interval in which the solution of (2.1) with initial value α exists. For

simplicity in the statements, we shall assume hereafter that $\varphi_t(\alpha)$ is always defined.

Flows possess the *group property*: φ_0 is the identity map in $\mathbb{R}^D$ and, for arbitrary real s and t,

$$\varphi_t \circ \varphi_s = \varphi_{s+t}. \tag{2.6}$$

In particular, for each t,

$$(\varphi_t)^{-1} = \varphi_{-t}, \tag{2.7}$$

that is, φ_{-t} is the inverse of the map φ_t. For the harmonic oscillator example, the group property simply states that a rotation of angle s followed by a rotation of angle t has the same effect as a rotation of angle $s + t$.

2.2. Hamiltonian systems

The Hamiltonian formalism (Marsden and Ratiu 1999, Arnol'd 1989) is essential for understanding HMC algorithms.

2.2.1. Hamiltonian vector fields

Assume that the dimension D of (2.1) is even, $D = 2d$, and write $x = (q, p)$ with $q, p \in \mathbb{R}^d$. Then the system (2.1) is said to be *Hamiltonian* if there is a function $H : \mathbb{R}^{2d} \to \mathbb{R}$ such that, for $i = 1, \ldots, d$, the scalar components f^i of f are given by

$$f^i(q,p) = +\frac{\partial H}{\partial q^i}(q,p), \quad f^{d+i}(q,p) = -\frac{\partial H}{\partial p^i}(q,p).$$

Thus, the system is

$$\frac{\mathrm{d}q^i}{\mathrm{d}t} = +\frac{\partial H}{\partial q^i}(q,p), \quad \frac{\mathrm{d}p^i}{\mathrm{d}t} = -\frac{\partial H}{\partial p^i}(q,p),$$

or, in vector notation (Sanz-Serna and Calvo 1994),

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} q \\ p \end{bmatrix} = J^{-1}\nabla H(q,p), \tag{2.8}$$

where

$$\nabla H = \left[\frac{\partial H}{\partial q^1}, \ldots, \frac{\partial H}{\partial q^d}, \frac{\partial H}{\partial p^1}, \ldots, \frac{\partial H}{\partial p^d}\right]^T$$

and

$$J = \begin{bmatrix} 0_{d\times d} & -I_{d\times d} \\ I_{d\times d} & 0_{d\times d} \end{bmatrix}.$$

The function H is called the *Hamiltonian*, $\mathbb{R}^{2d}$ is the *phase space*, and d is the *number of degrees of freedom*.

A system of the special form (2.2) is Hamiltonian if and only if $F = -\nabla\mathcal{U}(q)$ for a suitable real-valued function $\mathcal{U}$, that is,

$$\frac{\mathrm{d}}{\mathrm{d}t}q = M^{-1}p, \quad \frac{\mathrm{d}}{\mathrm{d}t}p = -\nabla\mathcal{U}(q); \tag{2.9}$$

when that is the case,

$$H(q,p) = \mathcal{T}(p) + \mathcal{U}(q), \quad \text{with } \mathcal{T}(p) = \frac{1}{2}p^T M^{-1}p. \tag{2.10}$$

In applications to mechanics, $\mathcal{T}$ and $\mathcal{U}$ are respectively the *potential* and *kinetic* energy and H represents the *total energy* in the system. The harmonic oscillator (2.3) provides the simplest example; there $\mathcal{T} = (1/2)p^2$, $\mathcal{U} = (1/2)q^2$.

2.2.2. Symplecticness and preservation of oriented volume

A mapping $\Phi : \mathbb{R}^{2d} \to \mathbb{R}^{2d}$ is said to be *symplectic* or *canonical* if, at each point $(q,p) \in \mathbb{R}^{2d}$,

$$\Phi'(q,p)^T J \Phi'(q,p) = J \tag{2.11}$$

($\Phi'(q,p)$ denotes the $2d \times 2d$ Jacobian matrix of Φ). The (analytic) condition (2.11) has a geometric interpretation in terms of preservation of two-dimensional areas (Arnol'd 1989); this interpretation is not required to understand the rest of the paper.

When $d = 1$, if we set $\Phi(q,p) = (q^*,p^*)$, the condition (2.11), after multiplying the matrices on the left-hand side, is seen to be equivalent to

$$\frac{\partial q^*}{\partial q}\frac{\partial p^*}{\partial p} - \frac{\partial q^*}{\partial p}\frac{\partial p^*}{\partial q} = 1.$$

The left-hand side is the Jacobian determinant of Φ and therefore the transformation Φ is symplectic if and only if the mapping $(q,p) \mapsto (q^*,p^*)$ preserves oriented area in the (q,p)-plane, that is, for any domain D the oriented area of the image $\Phi(D) \subset \mathbb{R}^2$ coincides with the oriented area of D.[1] For instance, for each t, the rotation in (2.5) is a symplectic transformation in $\mathbb{R}^2$.

For general d the following result holds (Arnol'd 1989, Section 38).

Proposition 2.2. For a symplectic transformation the determinant of Φ' equals 1. Therefore symplectic transformations preserve the oriented volume in $\mathbb{R}^{2d}$, that is, $\Phi(D)$ and D have the same oriented volume for each domain $D \subset \mathbb{R}^{2d}$.

[1] Preservation of oriented area means that D and $\Phi(D)$ have the same orientation and (two-dimensional Lebesgue) measure. The transformation (symmetry) $(q,p) \mapsto (q,-p)$ preserves measure but not oriented area.

For $d > 1$, preservation of oriented volume is a strictly weaker property than symplecticness.

The proof of the following two results is easy using (2.11).

Proposition 2.3. The composition $\Phi_1 \circ \Phi_2$ of two symplectic mappings is itself symplectic.

Proposition 2.4. The change of variables $(q, p) = \Phi(\bar{q}, \bar{p})$ with Φ symplectic transforms the Hamiltonian system of differential equations (2.8) into a system for $(\bar{q}, \bar{p})$ that is also Hamiltonian. Moreover, the Hamiltonian function $\bar{H}$ of the transformed system is the result of changing variables in H, that is, $\bar{H} = H \circ \Phi$.

In view of the following important general result (Marsden and Ratiu 1999, Proposition 2.6.2), the symplecticness of the rotation (2.5) noted above is a manifestation of the Hamiltonian character of the harmonic oscillator.

Theorem 2.5. Let $D = 2d$. The system (2.1) with flow φ_t is Hamiltonian if and only if, for each real t, φ_t is a symplectic mapping.

Thus symplecticness is a characteristic property that allows us to decide whether a differential system is Hamiltonian or otherwise in terms of its *flow*, without knowing the *vector field* (right-hand side) f of the equation.

We recall that a flow preserves oriented volume if and only if the corresponding vector field f is divergence-free ($\nabla \cdot f = \sum_i (\partial/\partial x^i) f^i = 0$). If $d > 1$, there are divergence-free differential systems in $\mathbb{R}^{2d}$ that are not Hamiltonian; their flows preserve oriented volume but are not symplectic.

The behaviour of the solutions of Hamiltonian problems is very different from that encountered in 'general' systems; some features that are 'the rule' in Hamiltonian systems are exceptional in non-Hamiltonian systems. Such special behaviour of Hamiltonian solutions may always be traced back to the symplecticness of the flow. As a very simple example, we consider once more the harmonic oscillator (2.3). The origin is a centre: a neutrally stable equilibrium surrounded by periodic trajectories. Small perturbations of the right-hand side of (2.3) generically destroy the centre; after perturbation the trajectories become spirals and the origin becomes either an asymptotically stable node (trajectories spiral in) or an unstable node (trajectories spiral out). However, if the perturbation is such that the perturbed system is also Hamiltonian, then the centre will not disappear under small perturbations.

2.2.3. Preservation of energy

If $(q(t), p(t))$ is a solution of (2.8), then

$$\frac{\mathrm{d}}{\mathrm{d}t} H(q(t), p(t)) = \nabla H(q(t), p(t))^T J^{-1} \nabla H(q(t), p(t)) = 0,$$

because J^{-1} is skew-symmetric. Therefore we may state the following.

Theorem 2.6. The value of the Hamiltonian function is preserved by the flow of the corresponding Hamiltonian system, that is, $H \circ \varphi_t = H$ for each real t.

In applications to the physical sciences, this result is usually the mathematical expression of the principle of conservation of energy. Unlike symplecticness, which is a characteristic property, conservation of energy on its own does not ensure that the underlying system is Hamiltonian. There are many examples of non-Hamiltonian systems whose solutions preserve the value of a suitable energy function.

2.2.4. Preservation of the canonical probability measure

Let β denote a positive constant and assume that H is such that

$$Z = \int_{\mathbb{R}^{2d}} \exp(-\beta H(q,p))\,\mathrm{d}q\,\mathrm{d}p < \infty.$$

Then we have the following result, which is a direct consequence of the fact that φ_t preserves both the volume element $\mathrm{d}q\,\mathrm{d}p$ (Proposition 2.2) and the value of $\exp(-\beta H)$ (because, according to Theorem 2.6, it preserves the value of H).

Theorem 2.7. The probability measure μ in $\mathbb{R}^{2d}$ with density (with respect to the ordinary Lebesgue measure) $Z^{-1}\exp(-\beta H(q,p))$ is preserved by the flow of the Hamiltonian system (2.8), that is, $\mu(\varphi_t(D)) = \mu(D)$ for each domain $D \subset \mathbb{R}^{2d}$ and each real t.

In statistical mechanics (Allen and Tildesley 1987, Frenkel and Smit 2002, Tuckerman 2010), if (2.8) describes the dynamics of a physical system and β is the inverse of the absolute temperature, then μ is the *canonical* measure that governs the distribution of (q,p) over an ensemble of many copies of the given system when the system is in contact with a heat bath at constant temperature, that is, $Z^{-1}\exp(-\beta H(q,p))\,\mathrm{d}q\,\mathrm{d}p$ represents the fraction of copies with momenta between p and $p + \mathrm{d}p$, and configuration between q and $q + \mathrm{d}q$. Note that under the canonical distribution, (local) minima of the energy H correspond to (local) maxima of the probability density function, that is, to *modes* of the distribution. Also, if the temperature decreases (β increases), it is less likely to find the system at a location (q,p) with high energy.

For Hamiltonian functions of the particular form in (2.10), we may factorize

$$\exp(-\beta H(q,p)) = \exp\left(-\frac{1}{2}\beta\, p^T M^{-1} p\right) \times \exp(-\beta\, \mathcal{U}(q))$$

and therefore, under the canonical distribution, q and p are stochastically independent. The (marginal) distribution of the configuration variables q has probability density function proportional to $\exp(-\beta\mathcal{U}(q))$. The momenta p possess a Gaussian distribution with zero mean and covariance matrix equal to M. These distributions are associated with the names of Boltzmann, Gibbs and Maxwell. Hereafter we refer to the canonical measure as the Boltzmann–Gibbs distribution.

2.3. Reversible systems

Assume now that S is a linear *involution* in $\mathbb{R}^D$, that is, a linear map such that $S(S(x)) = x$ for each x. A mapping $\Phi : \mathbb{R}^D \to \mathbb{R}^D$ is said to be *reversible* (with respect to S) if, for each x, $S(\Phi(x)) = \Phi^{-1}(S(x))$ or, more compactly,

$$S \circ \Phi = \Phi^{-1} \circ S. \tag{2.12}$$

The following results have easy proofs.

Proposition 2.8. If Φ is S reversible, then

$$|\det \Phi'(S(\Phi(x)))| = |\det \Phi'(x)|^{-1},$$

for each x.

Proposition 2.9. If Φ_1 is S-reversible, then $\Phi_1 \circ \Phi_1$ is S-reversible. If Φ_1 and Φ_2 are S-reversible, then the symmetric composition $\Phi_1 \circ \Phi_2 \circ \Phi_1$ is S-reversible.

Theorem 2.10. Consider the system (2.1) with flow φ_t. The following statements are equivalent.

- For each t, φ_t is an S-reversible mapping.
- For each $x \in \mathbb{R}^D$, $S(f(x)) = -f(S(x))$, that is, $S \circ f = -f \circ S$.

Systems of differential equations that satisfy the conditions in the theorem are said to be *reversible* (with respect to S). Systems of the particular form (2.2) are reversible with respect to the *momentum flip* involution

$$S(q, p) = (q, -p). \tag{2.13}$$

If (2.2) describes a mechanical system, then (2.12) expresses the well-known time-reversibility of mechanics: if (q_{in}, p_{in}) is the initial state of a system and (q_f, p_f) the final state after t units of time have elapsed, then the state $(q_f, -p_f)$ evolves in t units of time to the state $(q_{in}, -p_{in})$.

Proposition 2.11. The Hamiltonian system (2.8) is reversible with respect to the momentum flip involution (2.13) if and only if H is an even function of p, that is, $H(q, -p) = H(q, p)$ for all q and p.

Figure 6.1 (see page 160) illustrates the reversibility of the Hamiltonian flow corresponding to a one-degree-of-freedom double-well potential.

As is the case for Hamiltonian systems, reversible systems have flows with special geometric properties, not shared by 'general' systems (Lamb and Roberts 1998).

2.4. The Lie bracket

If $\varphi_t^{(f)}$ and $\varphi_t^{(g)}$ denote respectively the flows of the D-dimensional systems

$$\frac{\mathrm{d}}{\mathrm{d}t}x = f(x), \quad \frac{\mathrm{d}}{\mathrm{d}t}x = g(x),$$

in general $\varphi_s^{(g)} \circ \varphi_t^{(f)} \neq \varphi_t^{(f)} \circ \varphi_s^{(g)}$; in fact, a Taylor expansion shows that, as t, s approach 0,

$$\varphi_s^{(g)}(\varphi_t^{(f)}(x)) - \varphi_t^{(f)}(\varphi_s^{(g)}(x)) = st\,[f,g](x) + O(t^3 + s^3),$$

where the *Lie bracket* (or *Lie–Jacobi* bracket or commutator) $[f,g]$ of f and g (Arnol'd 1989, Section 39) is the mapping $\mathbb{R}^D \to \mathbb{R}^D$ that at $x \in \mathbb{R}^D$ takes the value

$$[f,g](x) = g'(x)f(x) - f'(x)g(x). \tag{2.14}$$

Thus the magnitude of $[f,g]$ measures the lack of commutativity of the corresponding flows. The following result holds.

Theorem 2.12. With the preceding notation, $\varphi_s^{(g)} \circ \varphi_t^{(f)} = \varphi_t^{(f)} \circ \varphi_s^{(g)}$ for arbitrary t and s if and only if the Lie bracket $[f,g](x)$ vanishes at each $x \in \mathbb{R}^D$. When these conditions hold, we say that f and g commute.

For commuting f and g, $\varphi_t^{(g)} \circ \varphi_t^{(f)} = \varphi_t^{(f)} \circ \varphi_s^{(g)}$ provides the t flow of the system

$$\frac{\mathrm{d}}{\mathrm{d}t}x = f(x) + g(x).$$

The Lie bracket is *skew-symmetric*: $[f,g] = -[g,f]$ for arbitrary f and g. In addition, it satisfies the *Jacobi identity*,

$$[f_1,[f_2,f_3]] + [f_2,[f_3,f_1]] + [f_3,[f_1,f_2]] = 0,$$

for any f_1, f_2, f_3. In this way the vector space of all vector fields together with the operation $[\cdot,\cdot]$ is a *Lie algebra*.

For *Hamiltonian* systems it is possible to work in terms of the so-called Poisson bracket of the Hamiltonian functions (Arnol'd 1989, Section 40), rather than in terms of the Lie bracket of the fields.

Theorem 2.13. If the fields f and g are Hamiltonian with Hamiltonian functions H and K respectively, that is, $f = J^{-1}\nabla H$, $g = J^{-1}\nabla K$, then $[f,g]$ is also a Hamiltonian vector field. Moreover, the Hamiltonian function

of $[f,g]$ is given by $-\{H,K\}$,[2] where $\{H,K\}$ is the *Poisson bracket* of the functions H and K, defined as

$$\{H,K\} = (\nabla H)^T J^{-1} \nabla K.$$

For real-valued functions in $\mathbb{R}^{2d}$, the Poisson bracket operation is skew-symmetric, that is, $\{H,K\} = -\{K,H\}$, and satisfies the Jacobi identity

$$\{H_1,\{H_2,H_3\}\} + \{H_2,\{H_3,H_1\}\} + \{H_3,\{H_1,H_2\}\} = 0.$$

Let f and g be *reversible* vector fields. Differentiation in $g(S(x)) = -S(g(x))$ implies for the Jacobian that $g'(S(x))S = -S(g'(x))$, and it follows that $g'(S(x))f(S(x))) = S(g'(x)f(x))$ (no minus sign!). Then the Lie bracket of two reversible fields is *not* reversible, but rather satisfies the following property.

Proposition 2.14. If two vector fields f and g are S-reversible, then

$$[f,g](S(x)) = S([f,g](x)).$$

For three S-reversible vector fields, the iterated commutator $[f_1,[f_2,f_3]]$ is S-reversible.

3. Integrators

In the sampling algorithms studied later, differential systems like (2.1) or (2.2) have to be numerically integrated. In this section we review the required material. Hairer, Nørsett and Wanner (1993), Hairer and Wanner (2010) and Butcher (2016) provide extensive, authoritative treatises on the subject. A more concise introduction is given by Griffiths and Higham (2010). We begin by recalling some basic definitions, and later focus on splitting integrators and fixed h stability, as both play important roles in the implementation of HMC algorithms.

3.1. Preliminaries

Each *one-step numerical method or one-step integrator* for (2.1) is described by a map $\psi_h : \mathbb{R}^D \to \mathbb{R}^D$ that depends on a real parameter h, the *step size*. Given the initial value α, and a value of h ($h \neq 0$), the integrator generates a numerical trajectory, $x_0, x_1, x_2, \ldots$, defined by $x_0 = \alpha$ and, iteratively,

$$x_{n+1} = \psi_h(x_n), \quad n = 0,1,2,\ldots. \tag{3.1}$$

To compute x_{n+1} when x_n has already been found is to perform a (time) *step*. For each n, the vector x_n is an approximation to the value at time

[2] The minus sign here could be avoided by reversing the sign in the definition of the Poisson bracket. The definition of $\{\cdot,\cdot\}$ used here is the one traditionally used in mechanics.

$t_n = nh$ of the solution $x(t)$ of (2.1) with initial condition $x(0) = \alpha$, *i.e.* to $\varphi_{t_n}(\alpha)$. Typically h is positive and then the integration is forward in time $0 = t_0 < t_1 < t_2 < \cdots$, but in some applications it may be of interest to use $h < 0$ so as to get $0 = t_0 > t_1 > t_2 > \cdots$.

The simplest and best known integrator, *Euler's rule*, with

$$x_{n+1} = x_n + hf(x_n), \tag{3.2}$$

corresponds to the mapping $\psi_h(x) = x + hf(x)$. It uses one evaluation of f per step. *Explicit s-stage Runge–Kutta* formulas use s evaluations of f per step, $s = 1, 2, \ldots$, and are therefore s times more expensive *per step* than Euler's rule; examples include Runge's method

$$\psi_h(x) = x + hf\left(x + \frac{h}{2}f(x)\right)$$

(with two stages), several well-known formulas of Kutta with four stages[3] and the formulas within the popular MATLAB function ode45. A method with s stages will be competitive with Euler's rule only if it gives more accurate approximations than Euler's rule when this is operated with a step size s times shorter, so as to equalize computational costs.

Implicit Runge–Kutta integrators are also used in practice, and then ψ_h is defined by means of algebraic equations. For instance, the *midpoint* rule has

$$\psi_h(x) = x + hf\left(\frac{1}{2}(x + \psi_h(x))\right),$$

that is,

$$x_{n+1} = x_n + hf\left(\frac{1}{2}(x_n + x_{n+1})\right). \tag{3.3}$$

Here, computing x_{n+1} when x_n is already known requires us to solve a system of algebraic equations in $\mathbb{R}^D$. There are many more useful examples of implicit Runge–Kutta methods, including the so-called *Gauss* methods.

Remark 3.1. Note that in the formulas displayed above h and f do not appear separately, but always in the combination hf. This clearly implies that, if, for a given method, x_n, $n = 0, 1, 2\ldots$ is a numerical trajectory with step size h corresponding to the system (2.1), it is also a numerical trajectory with step size λh for the system $(\mathrm{d}/\mathrm{d}t)x = \lambda^{-1}f(x)$ ($\lambda \neq 0$ denotes a constant). *We shall always assume that we deal with integrators having this property.*

[3] One of these formulas is known in some circles as *the* Runge–Kutta method; this terminology should be avoided as there are infinitely many Runge–Kutta methods.

A one-step integrator is called *symmetric* or *self-adjoint*[4] if

$$(\psi_h)^{-1} = \psi_{-h}, \tag{3.4}$$

so as to mimic the property (2.7) of the exact solution flow. The midpoint rule (3.3) provides an example. Explicit Runge–Kutta methods are never symmetric.

Multistep integrators, including the well-known Adams formulas, where the computation of x_{n+1} requires knowledge of $k \geq 2$ past values $x_n, x_{n-1}, \ldots, x_{n-k+1}$, may be very efficient, but will not be considered in this paper; they have seldom been applied in the context of sampling algorithms.

3.2. Order

Since, for the exact solution, the sequence $x(0) = \alpha, x(t_1), x(t_2), \ldots$ satisfies $x(t_{n+1}) = \varphi_h(x(t_n))$, $n = 0, 1, \ldots$, rather than (3.1), for the numerical integrator to make sense it is necessary that ψ_h be an approximation to φ_h. The integrator is said to be *consistent* if, at each fixed $x \in \mathbb{R}^D$, $\psi_h(x) - \varphi_h(x) = O(h^2)$ as $h \to 0$. If $\psi_h(x) - \varphi_h(x) = O(h^{\nu+1})$, ν a positive integer, then the integrator is (consistent) of *order* $\geq \nu$. A method of order $\geq \nu$ that is not of order $\geq \nu + 1$ is said to have order ν. Euler's rule (3.2) has order 1, the four-stage formulas of Kutta have order 4 and the midpoint rule (3.3) has order 2. We shall see later (Theorem 4.7) that the order of a symmetric integrator is an *even* integer. All integrators to be considered hereafter are assumed to be consistent.

The vector $\psi_h(x) - \varphi_h(x)$ is called the *local error* at x: it is the difference between the result of a single time-step of the numerical method starting from x and the result of the application of the h-flow to the *same* point x. For future reference, we note that the expansion of the local error for Euler's method is given by

$$\begin{aligned}\psi_h(x) - \varphi_h(x) &= (x + hf(x)) - \left(x + hf(x) + \frac{h^2}{2} f'(x)f(x) + O(h^3)\right) \\ &= -\frac{h^2}{2} f'(x)f(x) + O(h^3).\end{aligned} \tag{3.5}$$

In the expansion of the true solution flow we have used that $(\mathrm{d}/\mathrm{d}t)x = f(x)$ and $(\mathrm{d}^2/\mathrm{d}t^2)x = f'(x)(\mathrm{d}/\mathrm{d}t)x = f'(x)f(x)$.

The local error does not *per se* give information on the *global error* at t_n, *i.e.* on the difference $x_n - x(t_n)$. This is because $x_n = \psi_h(x_{n-1})$ while $x(t_n)$ is the result of the application of the h-flow to $x(t_{n-1})$: ψ_h and φ_h are *not*

[4] Even though Sanz-Serna and Calvo (1994) or Hairer, Lubich and Wanner (2010) use the term 'self-adjoint', there are reasons to avoid that terminology (Sanz-Serna 2016).

applied to the *same* point.[5] However, the following result holds.

Theorem 3.2. If the (one-step) integrator is consistent of order ν, then, for each fixed initial value $x_0 = x(0)$ and $T > 0$,

$$\max_{0 \le t_n \le T} |x_n - x(t_n)| = O(h^\nu), \quad h \to 0+. \tag{3.6}$$

This expresses the fact that the integrator is *convergent* of order ν. Of course, the corresponding result holds if the integration is carried out backward in time over $-T \le t \le 0$.

Remark 3.3. The global errors $x_n - x(t_n)$, in addition to possessing an $O(h^\nu)$ bound as above, have an *asymptotic expansion* in powers of h. For instance, if we restrict the attention to the leading h^ν-term in the expansion, we have

$$x_n - x(t_n) = h^\nu a(x(0), t_n) + h^\nu r(x(0), t_n; h),$$

where the function a is independent of h and, for each fixed initial value $x(0)$, $\max_{0 \le t \le T} |r(x(0), t, h)|$ tends to 0 as $h \to 0+$. Thus, for h sufficiently small, the global error is approximately equal to $h^\nu a(x(0), t_n)$: halving h divides the error by a factor 2^ν.

Remark 3.4. In the description above, the step points t_n are uniformly spaced. General-purpose software uses *variable time-steps*: $t_{n+1} = t_n + h_n$ where h_n changes with n. For each n, the value of h_n is chosen by the algorithm to ensure that the local error at x_n is below a user-prescribed tolerance. Since, for reasons to be explained in Section 4, HMC integrations are carried out with constant step size, we shall not concern ourselves with variable step sizes.

3.3. Splitting methods

Euler's method and the other integrators mentioned above may be applied to any given system (2.1). Other techniques only make sense for *particular* classes of systems and cannot therefore be incorporated into general-purpose software; however, they have gained much popularity in recent decades due, among other things, to their role in *geometric integration* (see Section 4). Of special importance to us is the class of *splitting* integrators that we consider next. Blanes and Casas (2016) provide a very good source of information.

[5] In order to bound the global error in terms of the local error, that is, to obtain convergence from consistency, a *stability* property is needed. Hence the well-known slogan 'stability + consistency imply convergence'. The one-step integrators considered here always have the required stability and therefore, for them, consistency implies convergence as in Theorem 3.2. The concept of stability that features in the slogan is *different* from the concept of fixed h stability studied in Section 3.4 (Sanz-Serna 1991).

3.3.1. The Lie–Trotter formula

Splitting methods are applicable to cases where the right-hand side of (2.1) may be split into two parts,

$$\frac{\mathrm{d}}{\mathrm{d}t}x = f(x) = f^{(\mathrm{A})}(x) + f^{(\mathrm{B})}(x), \tag{3.7}$$

in such a way that the flows $\varphi_t^{(\mathrm{A})}$ and $\varphi_t^{(\mathrm{B})}$ of the *split systems*

$$\frac{\mathrm{d}}{\mathrm{d}t}x = f^{(\mathrm{A})}(x), \quad \frac{\mathrm{d}}{\mathrm{d}t}x = f^{(\mathrm{B})}(x), \tag{3.8}$$

are available analytically. To avoid trivial cases, we shall hereafter assume that the Lie bracket $[f^{(\mathrm{A})}, f^{(\mathrm{B})}]$ does not vanish identically, because otherwise Theorem 2.12 shows that (3.7) may also be solved analytically without resorting to numerical approximations.

Systems of the particular form (2.2) provide an important example. When they are split by taking

$$(\mathrm{A}) \qquad \frac{\mathrm{d}}{\mathrm{d}t}q = M^{-1}p, \quad \frac{\mathrm{d}}{\mathrm{d}t}p = 0 \tag{3.9}$$

and

$$(\mathrm{B}) \qquad \frac{\mathrm{d}}{\mathrm{d}t}q = 0, \quad \frac{\mathrm{d}}{\mathrm{d}t}p = F(q), \tag{3.10}$$

the flows are explicitly given by

$$\varphi_t^{(\mathrm{A})}(q,p) = (q + tM^{-1}p, p)$$

and

$$\varphi_t^{(\mathrm{B})}(q,p) = (q, p + tF(q)).$$

In molecular dynamics (Schlick 2002) these mappings are respectively called a *drift* (q advances with constant speed and the momentum p remains constant) and a *kick* (the system stays in its current configuration, and the momentum is incremented by the action of the force). A direct computation of the Lie bracket shows that, except in the trivial case where F vanishes identically, the vector fields in (3.9)–(3.10) do not commute.

A simple Taylor expansion proves that the Lie–Trotter formula

$$\psi_h = \varphi_h^{(\mathrm{B})} \circ \varphi_h^{(\mathrm{A})} \tag{3.11}$$

defines a first-order integrator for (3.7):

$$\psi_h(x) - \varphi_h(x) = \frac{h^2}{2}[f^{(\mathrm{A})}, f^{(\mathrm{B})}](x) + O(h^3). \tag{3.12}$$

(Observe that, not unexpectedly, the coefficient of the leading power of h is

proportional to the Lie bracket.) While $f^{(\mathrm{A})}$ and $f^{(\mathrm{B})}$ contribute simultaneously to the change of x in (3.7), they do so successively in the Lie–Trotter integrator. By swapping the roles of A and B, we have the alternative integrator

$$\psi_h = \varphi_h^{(\mathrm{A})} \circ \varphi_h^{(\mathrm{B})}.$$

3.3.2. Strang's formula

The most popular splitting integrator for (3.7) corresponds to Strang's formula (Strang 1963)

$$\psi_h = \varphi_{(1/2)h}^{(\mathrm{B})} \circ \varphi_h^{(\mathrm{A})} \circ \varphi_{(1/2)h}^{(\mathrm{B})} \tag{3.13}$$

and, as a Taylor expansion shows, has second-order accuracy, $\psi_h(x) - \varphi_h(x) = O(h^2)$ as $h \to 0$. More precisely,

$$\begin{aligned}\psi_h(x) - \varphi_h(x) = {} & \frac{h^3}{12}[f^{(\mathrm{A})}, [f^{(\mathrm{A})}, f^{(\mathrm{B})}]](x) \\ & + \frac{h^3}{24}[f^{(\mathrm{B})}, [f^{(\mathrm{A})}, f^{(\mathrm{B})}]](x) + O(h^4),\end{aligned} \tag{3.14}$$

so that the leading term of the local error is a linear combination of two so-called *iterated Lie brackets*.

When applied to the particular case (2.2), the formula (3.13) yields the well-known *velocity Verlet* integrator, the method of choice in molecular dynamics; the step $n \to n+1$ comprises two kicks of duration $h/2$ separated by a drift of duration h:

$$\begin{aligned} p_{n+1/2} &= p_n + \frac{h}{2} F(q_n), \\ q_{n+1} &= q_n + hM^{-1} p_{n+1/2}, \\ p_{n+1} &= p_{n+1/2} + \frac{h}{2} F(q_{n+1}). \end{aligned}$$

The evaluations of F represent the bulk of the computational cost of the algorithm. The value $F(q_{n+1})$ to be used in the first kick of the next step, $n+1 \to n+2$, is the same used in the second kick of the current step. In this way, while the very first step requires two evaluations of F, subsequent steps only need one.

When N steps of the method (3.13) are taken, the map that advances the numerical solution from x_0 to x_N, that is,

$$\psi_h^N = \overbrace{\left(\varphi_{(1/2)h}^{(\mathrm{B})} \circ \varphi_h^{(\mathrm{A})} \circ \varphi_{(1/2)h}^{(\mathrm{B})}\right) \circ \cdots \circ \left(\varphi_{(1/2)h}^{(\mathrm{B})} \circ \varphi_h^{(\mathrm{A})} \circ \varphi_{(1/2)h}^{(\mathrm{B})}\right)}^{N \text{ times}},$$

may be rewritten with the help of the group property (2.6) in the *leapfrog*

form

$$\psi_h^N = \varphi^{(\mathrm{B})}_{(1/2)h} \circ \overbrace{\left(\varphi_h^{(\mathrm{A})} \circ \varphi_h^{(\mathrm{B})}\right) \circ \cdots \circ \left(\varphi_h^{(\mathrm{A})} \circ \varphi_h^{(\mathrm{B})}\right)}^{N-1 \text{ times}} \circ \varphi_h^{(\mathrm{A})} \circ \varphi^{(\mathrm{B})}_{(1/2)h};$$

now the right-hand side only uses $N+1$ times the flow $\varphi_t^{(\mathrm{B})}$. In the particular case (3.9)–(3.10), the combination $\varphi_h^{(\mathrm{A})} \circ \varphi_h^{(\mathrm{B})}$ corresponds to the following formulas to advance the numerical solution, $n = 1, \ldots, N-1$:

$$\begin{aligned} p_{n+1/2} &= p_{n-1/2} + hF(q_n), \\ q_{n+1} &= q_n + hM^{-1}p_{n+1/2}; \end{aligned}$$

p jumps over q and then q jumps over p like children playing leapfrog. The leapfrog implementation makes apparent the truth of an earlier observation: N steps of the velocity Verlet integrator may be implemented with $N+1$ evaluations of F.

Strang's method is symmetric:

$$\begin{aligned} (\psi_h)^{-1} &= \left(\varphi^{(\mathrm{B})}_{(1/2)h}\right)^{-1} \circ \left(\varphi_h^{(\mathrm{A})}\right)^{-1} \circ \left(\varphi^{(\mathrm{B})}_{(1/2)h}\right)^{-1} \\ &= \varphi^{(\mathrm{B})}_{-(1/2)h} \circ \varphi^{(\mathrm{A})}_{-h} \circ \varphi^{(\mathrm{B})}_{-(1/2)h} \\ &= \psi_{-h}. \end{aligned}$$

It is clear that the symmetry is a consequence of the *palindromic* structure of (3.13), that is, the formula reads the same from left to right as from right to left.

The roles of A and B in (3.13) may be interchanged:

$$\psi_h = \varphi^{(\mathrm{A})}_{(1/2)h} \circ \varphi^{(\mathrm{B})}_h \circ \varphi^{(\mathrm{A})}_{(1/2)h}. \tag{3.15}$$

For (3.9)–(3.10) one then obtains the *position Verlet* integrator: one step comprises two drifts of duration $h/2$ and one kick of duration h.

3.3.3. Splitting formulas with more stages

It is of course possible to use splitting formulas more sophisticated than Strang's. For instance, for any choice of the real parameter b, we may consider the integrator

$$\psi_h = \varphi^{(\mathrm{B})}_{bh} \circ \varphi^{(\mathrm{A})}_{(1/2)h} \circ \varphi^{(\mathrm{B})}_{(1-2b)h} \circ \varphi^{(\mathrm{A})}_{(1/2)h} \circ \varphi^{(\mathrm{B})}_{bh}, \tag{3.16}$$

where we observe that, in one step, the A and B flows of the systems (3.8) act for a total duration of h units of time each to ensure consistency. Due to its palindromic structure, this integrator is symmetric, and from Theorem 4.7 its order is at least two. It turns out that the order is exactly two for all choices of b (Blanes, Casas and Sanz-Serna 2014). The order of splitting integrators is discussed later in relation to the concept of modified equations.

Even though three B flows and two A flows feature in (3.16), N steps of the integrator only require the computation of $2N + 1$ B flows and $2N$ A flows; this is seen by combining flows as we did above for the Strang case. We say that (3.16) is a *palindromic two-stage* integrator.[6] The method (3.16) may be denoted by

$$(b, 1/2, (1 - 2b), 1/2, b). \tag{3.17}$$

Similarly, one may consider the two-parameter family of *palindromic three-stage* splittings

$$(b, a, 1/2 - b, 1 - 2a, 1/2 - b, a, b). \tag{3.18}$$

A full description of this family is given by Campos and Sanz-Serna (2017), who suggest parameter choices for various applications. There is a unique choice of a and b resulting in a fourth-order method often associated with Yoshida's name (Yoshida 1990); for all other choices, the order is $\nu = 2$.

The family of *palindromic s-stage* splitting formulas is given by

$$(b_1, a_1, b_2, a_2, \ldots, a_{s'}, b_{s'+1}, a_{s'}, \ldots, a_2, b_2, a_1, b_1) \tag{3.19}$$

if $s = 2s'$ is even, and by

$$(b_1, a_1, b_2, a_2, \ldots, b_{s'}, a_{s'}, b_{s'}, \ldots, a_2, b_2, a_1, b_1) \tag{3.20}$$

if $s = 2s' - 1$. After imposing the consistency requirement that at each step the A and B flows act during h units of time each, the family has $s - 1$ parameters left. By taking s sufficiently high it is possible to achieve any desired order (Sanz-Serna and Calvo 1994, Section 13.1). Clearly, increasing the number of stages does *not* lead to integrators with more complicated implementation; regardless of the value of s, numerical integrations only consist of a sequence of flows of the split systems.

It is not necessary to add that to each of the integrators just described there corresponds a second integrator found by swapping the roles of the systems A and B, just as (3.15) corresponds to (3.13).

So far our attention has focused on palindromic formulas as these will be important later in connection with the property of reversibility (Theorem 4.2). It is also possible to consider splittings of the form

$$\psi_h = \varphi^{(\mathrm{B})}_{b_r h} \circ \varphi^{(\mathrm{A})}_{a_r h} \circ \cdots \circ \varphi^{(\mathrm{B})}_{b_2 h} \circ \varphi^{(\mathrm{A})}_{a_2 h} \circ \varphi^{(\mathrm{B})}_{b_1 h} \circ \varphi^{(\mathrm{A})}_{a_1 h},$$

which we denote as

$$(b_r, a_r, \ldots, b_2, a_2, b_1, a_1) \tag{3.21}$$

[6] But (3.16) is still a *one-step* integrator, because x_{n+1} is determined by x_n; for two-step schemes the computation of x_{n+1} requires knowledge of both x_n *and* x_{n-1}. The term 'stage' is borrowed from the Runge–Kutta literature.

($\sum_i a_i = \sum_i b_i = 1$). The palindromic formulas in (3.19)–(3.20) are particular instances of this general format because it is always possible to set $a_1 = 0$ in (3.21).

Remark 3.5. When a splitting integrator of the general form (3.21) is applied to the splitting (3.9)–(3.10) of the system (2.2), the result coincides with the application of a so-called symplectic, explicit Runge–Kutta–Nyström (RKN) method. The properties (order, stability, *etc.*) of such an integrator may therefore be studied either by using techniques pertaining to splitting integrators (as is done in this paper) or by employing an RKN approach. The second methodology was favoured in the early years of geometric integration (Sanz-Serna 1992, Sanz-Serna and Calvo 1994).

3.4. Fixed h stability

If two numerical schemes are candidates to integrate an initial value problem for a given system (2.1), then the scheme that leads to smaller global errors for a given computational cost may seem more desirable (but, in the context of geometric integration, the geometric properties may play a role when choosing the integrator: see Section 4 below). Even though global errors may be bounded as in (3.6), in practice it is almost always impossible, for the problem at hand, to estimate realistically the error constant implied in the $O(h^\nu)$ notation in the bound. For this reason, the literature on numerical integrators has traditionally resorted to well-chosen *model problems* where both the numerical and true solutions, x_n and $x(t_n)$, may be written down in closed form; the performance of the various integrators on the model problem may then be investigated analytically and is taken as an indication of their performance when applied to realistic problems. Note that the actual numerical solution of the model problem cannot be expected to be of real practical interest, since in that problem the true solution is available analytically.

3.4.1. One degree of freedom

For our purposes, the model problem of choice is the harmonic oscillator (2.3). For Runge–Kutta methods, for the splitting algorithms in Section 3.3 (and in fact for all one-step integrators of practical interest) a time-step $(q_{n+1}, p_{n+1}) = \psi_h(q_n, p_n)$ may be expressed as

$$\begin{bmatrix} q_{n+1} \\ p_{n+1} \end{bmatrix} = \tilde{M}_h \begin{bmatrix} q_n \\ p_n \end{bmatrix}, \quad \tilde{M}_h = \begin{bmatrix} A_h & B_h \\ C_h & D_h \end{bmatrix} \tag{3.22}$$

for suitable method-dependent coefficients A_h, B_h, C_h, D_h. The evolution over n time-steps is then given by

$$\begin{bmatrix} q_n \\ p_n \end{bmatrix} = \tilde{M}_h^n \begin{bmatrix} q_0 \\ p_0 \end{bmatrix}, \tag{3.23}$$

an expression to be compared with (2.4).

If a given $h > 0$ is such that $|\tilde{M}_h^n| \to \infty$ as $n \to \infty$, the magnitude of numerical solution (q_n, p_n) will grow unboundedly, while the true solution of course remains bounded as $t \to \infty$. Necessarily, global errors will be large for large n. Then the integrator is said to be *unstable* for that particular choice of h.

Example 3.6. For Euler's rule (3.2) we find

$$A_h = D_h = 1, \quad B_h = -C_h = h.$$

The eigenvalues of $\tilde{M}_h$ are $1 \pm \mathrm{i}h$ with modulus $(1 + h^2)^{1/2}$ and, therefore, for *any* fixed $h > 0$, the powers $\tilde{M}_h^n$ grow exponentially as n increases. Due to this numerical instability, Euler's rule is completely unsuitable for integrating the harmonic oscillator, and this rules it out as a method to integrate more complicated oscillatory problems.

More precisely, in the step $n \to n + 1$, the radius $r = (q^2 + p^2)^{1/2}$, which remains constant for the true solution, grows like

$$r_{n+1} = (1 + h^2)^{1/2} r_n = \left(1 + \frac{h^2}{2} + O(h^4)\right) r_n \tag{3.24}$$

for the Euler solution, so that $r_n = (1 + h^2)^{n/2} r_0$. Note that, taking limits as $n \to \infty$, $h \to 0$, with fixed nh, we find $r_n \to r_0 = r(0) = r(nh)$, as it corresponds to a convergent method.

Example 3.7. The midpoint rule (3.3) has

$$A_h = D_h = \frac{1 - h^2/4}{1 + h^2/4}, \quad B_h = -C_h = \frac{h}{1 + h^2/4}.$$

The characteristic equation of $\tilde{M}_h$ is

$$\lambda^2 - 2A_h\lambda + 1, \tag{3.25}$$

and therefore the product of the eigenvalues is 1. For $h \neq 0$, $|A_h| < 1$ and the matrix has a pair of complex conjugate eigenvalues of unit modulus. Then the powers $\tilde{M}_h^n$ remain bounded as n increases and the method is stable for any h.

Example 3.8. Let us next consider Strang's splitting (3.13), applied with the splitting (3.9)–(3.10), which yields the velocity Verlet algorithm. This has

$$A_h = D_h = 1 - h^2/2, \quad B_h = h, \quad C_h = -h + h^3/4;$$

the characteristic equation is again of the form (3.25). For $h > 2$, $A_h < -1$ and the eigenvalues are real and distinct, so that one of them has modulus > 1, and therefore the powers $\tilde{M}_h^n$ grow exponentially. For $h < 2$ the eigenvalues are complex of unit modulus and the powers $\tilde{M}_h^n$ remain bounded.

Table 3.1. Velocity Verlet integration of the harmonic oscillator. Relative errors after one or ten oscillation periods.

h	$t = T_{\text{per}}$	$t = 10\,T_{\text{per}}$
$T_{\text{per}}/4$	6.49×10^{-1}	2.00×10^{0}
$T_{\text{per}}/8$	1.60×10^{-1}	1.48×10^{0}
$T_{\text{per}}/16$	4.03×10^{-2}	4.00×10^{-1}
$T_{\text{per}}/32$	1.01×10^{-2}	1.01×10^{-1}

For $h = 2$, $\tilde{M}_h$ is a non-trivial Jordan block whose powers grow linearly (weak instability). To summarize, the integrator is unstable for $h \geq 2$ and has the *stability interval* $0 < h < 2$. Integrations with $h > 2$ lead to extremely large global errors, as we shall see below.

Example 3.9. For the alternative Strang formula (3.15), which yields the position Verlet algorithm, the coefficients are

$$A_h = D_h = 1 - h^2/2, \quad B_h = h - h^3/4, \quad C_h = -h.$$

The discussion is almost identical to the one in the preceding example. The stability interval is also $0 < h < 2$, as one may have guessed from the equal role that q and $-p$ play in the harmonic oscillator (2.3).

Example 3.10. In the situation of Example 3.8, are values of h below the upper limit 2 satisfactory? The answer of course depends on the accuracy required. For the initial condition $q = 1$, $p = 0$, and different stable values of h, Table 3.1 gives the relative error in the Euclidean norm

$$\frac{|(q_n - q(t_n), p_n - p(t_n))|}{|(q(t_n), p(t_n))|}$$

at the final integration time, when the integration is carried out over an interval of length either one oscillation period (second column) or ten oscillation periods (third column). Columns are consistent with the order of convergence as in (3.6). The last rows reveal that the error increases linearly with t (this will not be true when integrating other differential systems). In the first row the error grows more slowly than t: the numerical solution (q_n, p_n) remains close to the unit circle for all values of n and therefore errors cannot be substantially larger than the diameter of the circle. From the table we see that if we are interested in errors below 10% over an interval of length equal to ten oscillation periods, then h has to be taken below $2\pi/32 \approx 0.20$, that is, well below the upper end, $h = 2$, of the stability

interval. To provide an indication of the effect of using unstable values of h, we mention that with $h = \pi$ the error after one period is ≈ 46.4 and after ten periods $\approx 4.68 \times 10^{17}$.

Before we move to more complicated models, let us make two observations, valid for Runge–Kutta, splitting integrators and any other method of practical interest.

Remark 3.11. Because the problem is linear and rotationally invariant, the magnitude of the *relative* errors is independent of the initial condition $(q_0, p_0) \neq (0, 0)$.

Remark 3.12. Replacing the model (2.3) with the apparently more general system

$$\frac{\mathrm{d}q}{\mathrm{d}t} = \omega p, \quad \frac{\mathrm{d}p}{\mathrm{d}t} = -\omega q, \quad \omega > 0, \tag{3.26}$$

with oscillation period $2\pi/\omega$, does not really change things: in view of Remark 3.1, integrating (3.26) with step size h is equivalent to integrating (2.3) with step size h/ω. Because the length of the integration interval and the step size in Table 3.1 are given in terms of the oscillation period, the results displayed are valid for (3.26) for any choice of the value of ω. Regardless of the initial condition, if we are interested in relative errors below 10% over an interval of length $20\pi/\omega$ (equal to ten oscillation periods), then h has to be taken below $2\pi/(32\omega)$. Of course, for (3.26), stability requires that $h < 2/\omega$.

3.4.2. Several degrees of freedom; stability restrictions on h

Let us now move to the model with d degrees of freedom,

$$\frac{\mathrm{d}}{\mathrm{d}t}q = M^{-1}p, \quad \frac{\mathrm{d}}{\mathrm{d}t}p = -Kq,$$

where M and K are $d \times d$, symmetric positive-definite matrices. This is the Hamiltonian system with

$$H = \frac{1}{2}p^T M^{-1} p + \frac{1}{2}q^T K q. \tag{3.27}$$

In mechanics, M and K are the mass and stiffness matrices respectively.

This model may transformed into d uncoupled one-degree-of-freedom oscillators. In fact, factor $M = LL^T$ (which may be done in infinitely many ways) and diagonalize the symmetric positive-definite matrix $L^{-1}KL^{-T}$ as $U^T L^{-1} K L^{-T} U = \Omega^2$, with U orthogonal and Ω diagonal with diagonal entries ω_i, $i = 1, \ldots, d$. A simple computation yields the next result.

Proposition 3.13. With the notation above, the (non-canonical) change of dependent variables, $q = L^{-T}U\bar{q}$, $p = LU\Omega\bar{p}$, decouples the system into a collection of d harmonic oscillators:

$$\frac{\mathrm{d}\bar{q}^i}{\mathrm{d}t} = \omega_i \bar{p}^i, \quad \frac{\mathrm{d}\bar{p}^i}{\mathrm{d}t} = -\omega_i \bar{q}^i, \quad i = 1, \dots, d$$

(superscripts denote components).

Now, for all integrators of practical interest, *decoupling and numerical integration commute*: carrying out the integration in the old variables (q, p) yields the same result as successively (i) changing variables in the system, (ii) integrating each of the uncoupled oscillators, and (iii) translating the result to the old variables. *Therefore stability may be analysed under the assumption that the integration is performed in the uncoupled version.*

Example 3.14. Consider a particular case with $d = 2$, $M = L = I$, $\omega_1 = 1$ and $\omega_2 = 100$ and the initial condition

$$\bar{q}^1(0) = 1, \quad \bar{p}^1(0) = 0, \quad \bar{q}^2(0) = 0.01, \quad \bar{p}^2(0) = 0.$$

We wish to integrate with the velocity Verlet algorithm over $0 \le t \le 20\pi$ (ten periods of the slower oscillation) and aim at absolute errors of magnitude ≈ 0.1 in the Euclidean norm in the $\mathbb{R}^4$ space of the variables (q^1, q^2, p^1, p^2) or, equivalently, because here $q = U\bar{q}$ with U orthogonal, in the Euclidean norm of the variables $(\bar{q}^1, \bar{q}^2, \bar{p}^1, \bar{p}^2)$. On *accuracy* grounds it would be sufficient to take $h \approx 0.2$: from Example 3.10 we know this is enough to integrate the first uncoupled oscillator with the desired accuracy, and the second oscillator should not contribute significantly to the error, due to the smallness of $\bar{q}^2(t)$ and $\bar{p}^2(t)$ for all t. However, unless we take $h\omega_2 < 2$, *i.e.* $h < 0.02$, the errors in $\bar{q}^2(t)$ and $\bar{p}^2(t)$ will grow exponentially due to *instability*. In this example, and in many situations arising in practice, to avoid instabilities, the value of h has to be chosen much smaller than accuracy would require. As a result the computational effort to span the time interval of interest would be much higher than might be expected on accuracy grounds. These situations are called *stiff*. To deal with stiffness one may resort to suitable implicit integrators such as the midpoint rule, or explicit integrators with large stability intervals.

Remark 3.15. It may seem that the initial condition in the preceding example is somewhat contrived, as the sizes of two components of $\bar{q}$ are so unbalanced. This is not so: it is easily checked that, in terms of the original energy in (3.27), the contributions to H of the ω_1 and ω_2 oscillations are both equal to $1/2$. Those oscillations are called *normal modes* in mechanics.

The present discussion is continued in Section 4.4.

We emphasize that, while the preceding material refers to the quadratic Hamiltonian (3.27), one may expect that it has some relevance for Hamiltonians close to that model. However, there are cases where the behaviour of a numerical integrator departs considerably from its behaviour when applied to (3.27). This point is discussed in Section 9.2.

4. Geometric integration

Classically, the development of numerical integrators for ordinary differential equations focused on general-purpose methods (such as linear multistep or Runge–Kutta formulas) that were selected after analysing their local error and fixed h stability properties. Geometric integration (Sanz-Serna 1997) is a newer paradigm of numerical integration, where the interest lies in methods tailored to the problem at hand with a view to preserving some of its geometric features. The development of geometric integration started in the 1980s with the study of symplectic integrators for Hamiltonian systems by Feng Kang and others (Sanz-Serna 1992, Sanz-Serna and Calvo 1994). Useful monographs are those by Leimkuhler and Reich (2004), Hairer *et al.* (2010), Feng and Qin (2010) and Blanes and Casas (2016). We review the geometric integration of Hamiltonian and reversible systems and study the use of modified equations, a key tool for our purposes. We also examine the behaviour of geometric integrators in the harmonic model problem. The section concludes by showing the optimality of the stability interval of the Strang/Verlet integrator.

4.1. Hamiltonian problems

Splitting integrators are *symplectic* in the following sense. Assume that the system (3.7) is Hamiltonian and that it is split in such a way that both split systems (3.8) are also Hamiltonian. Then the splitting integrator mapping ψ_h is symplectic, as a composition (Proposition 2.3) of flows that are individually symplectic (Theorem 2.5). This ensures that the numerical solution shares the specific properties of Hamiltonian flows that derive from symplecticness. Note that for ψ_h to be symplectic it is not enough that the system being integrated is Hamiltonian; if the split vector fields $f^{(\mathrm{A})}$ and $f^{(\mathrm{B})}$ are not Hamiltonian themselves, then ψ_h cannot be expected to be symplectic. Of course, all possible splittings of a Hamiltonian vector field as a sum of Hamiltonian vector fields, $f = f^{(\mathrm{A})} + f^{(\mathrm{B})}$, may be obtained by splitting the Hamiltonian $H = H^{(\mathrm{A})} + H^{(\mathrm{B})}$ in an arbitrary way and then setting $f^{(\mathrm{A})} = J^{-1}\nabla H^{(\mathrm{A})}$, $f^{(\mathrm{B})} = J^{-1}\nabla H^{(\mathrm{B})}$. We sum up as follows.

Theorem 4.1. Assume that the Hamiltonian of the system (2.8) is written as $H = H^{(\mathrm{A})} + H^{(\mathrm{B})}$ and, correspondingly, split $f = J^{-1}\nabla H$ into $f^{(\mathrm{A})} = J^{-1}\nabla H^{(\mathrm{A})}$ and $f^{(\mathrm{B})} = J^{-1}\nabla H^{(\mathrm{B})}$. For any splitting integrator

(3.21) and any h, the mapping ψ_h is symplectic. In particular, ψ_h conserves oriented volume.

Note that the n-fold composition ψ_h^n that advances the numerical solution over n time-steps is then also symplectic (Proposition 2.3).

Some *implicit* Runge–Kutta methods, including the midpoint rule and Gauss methods, are also symplectic: ψ_h is a symplectic map whenever the system (2.1) being integrated is Hamiltonian as in (2.8). No explicit Runge–Kutta method is symplectic. In particular, Euler's rule is not symplectic; according to Example 3.6 the Euler discretization of the harmonic oscillator *increases* area, as $\tilde{M}_h$ has determinant $1+h^2$. This increase in area is related to the estimate (3.24) that shows that the Euler solution spirals outward, a non-Hamiltonian behaviour.

In the particular case of the Hamiltonian in (2.10), taking $\mathcal{T}$ and $\mathcal{U}$ to play the roles of $H^{(\mathrm{A})}$ and $H^{(\mathrm{B})}$ respectively leads to the splitting of the differential system given in (3.9)–(3.10) with $F = -\nabla\mathcal{U}$.

It would also be desirable to have integrators that preserved energy when applied to the Hamiltonian system (2.8), that is, $H \circ \psi_h = H$. Unfortunately, for realistic problems such a requirement is incompatible with ψ_h being symplectic (Sanz-Serna and Calvo 1994, Section 10.3). It is then standard practice to insist on symplecticness and sacrifice conservation of energy. There are several reasons for this. Symplecticness plays a key role in the Hamiltonian formalism (see Theorem 2.5). In addition, while, as we have seen, it is not difficult to find symplectic formulas, standard classes of integrators do not include energy-preserving schemes unless the energy is assumed to have particular forms. Finally, as we shall discuss below, symplectic schemes have small energy errors even when the integration interval is very long.

4.2. Reversible problems

For reversible systems we have, from Proposition 2.9 and Theorem 2.10, the following theorem.

Theorem 4.2. Assume that (3.7), and the split systems (3.8) are reversible with respect to the same involution S. If the system is integrated by means of a *palindromic* splitting integrator (3.19) or (3.20), then, for any h, the mapping ψ_h will also be reversible.

The midpoint rule and Gauss Runge–Kutta formulas also generate reversible mappings ψ_h whenever they are applied to a reversible system. No explicit Runge–Kutta method is reversible.

The n-fold composition ψ_h^n that advances the numerical solution over n time-steps is then also reversible (Proposition 2.9). However, note that if variable time-steps were taken, then the mapping $\psi_{h_n} \circ \cdots \circ \psi_{h_1}$, which

advances the solution from t_0 to t_{n+1}, would not be reversible. This is one of the reasons for not considering variable time-steps here; see Remark 3.4.

The use of reversible integrators (with constant step sizes) ensures that the numerical solution inherits relevant geometric properties of the true solution of the differential system (Cano and Sanz-Serna 1997, Cano and Sanz-Serna 1998).

4.3. Modified equations

Modified equations are rather old (see references in Griffiths and Sanz-Serna 1986); however, their use as a means to analyse numerical integrators has only become popular in the past twenty years, since the emergence of geometric integration (Sanz-Serna 1996).

4.3.1. Motivation

Let us first look at some examples.

Example 4.3. Suppose that the system (2.1) is solved with Euler's rule (3.2) and let ψ_h denote the corresponding map $x+hf(x)$. Consider the new system, parametrized by h,

$$\frac{\mathrm{d}}{\mathrm{d}t}x = \tilde{f}_h(x), \quad \tilde{f}_h(x) = f(x) - \frac{h}{2}f'(x)f(x), \tag{4.1}$$

with flow $\tilde{\varphi}_t$ (for simplicity, the dependence of this flow on the parameter h is not incorporated in the notation). By proceeding as in the derivation of (3.5), we find that the Taylor expansion of $\tilde{\varphi}_h$ in powers of h is

$$\begin{aligned}\tilde{\varphi}_h(x) &= x + h\tilde{f}_h(x) + \frac{h^2}{2}\tilde{f}_h'(x)\tilde{f}_h(x) + O(h^3)\\ &= x + h\left(f(x) - \frac{h}{2}f'(x)f(x)\right) + \frac{h^2}{2}f_h'(x)f_h(x) + O(h^3)\\ &= x + hf(x) + O(h^3).\end{aligned}$$

Thus, the Euler mapping $\psi_h(x)$, which differs from the flow of the system (2.1) being solved in $O(h^2)$ terms (first-order consistency), differs from the flow $\tilde{\varphi}_h$ of the so-called *modified or shadow system* (4.1) in $O(h^3)$ terms (second-order consistency). According to Theorem 3.2 (see Griffiths and Sanz-Serna 1986 for details), over bounded time intervals, the Euler solution differs from the corresponding solution of the modified system in $O(h^2)$ terms. Therefore we may expect that the properties of the Euler solutions for (2.1) will resemble the properties of the solutions of (4.1) more than they resemble the properties of solutions of (2.1) itself. When studying the properties of Euler's rule, working with (4.1) rather than with ψ_h should be advantageous, because differential equations are simpler to analyse than maps. An illustration is provided next.

Example 4.4. Let us particularize the preceding example to the case of the harmonic oscillator (2.3). The modified system (4.1) is found to be

$$\frac{\mathrm{d}q}{\mathrm{d}t} = p + \frac{h}{2}q, \quad \frac{\mathrm{d}p}{\mathrm{d}t} = -q + \frac{h}{2}p,$$

and from here a simple computation yields for the radius $r = (q^2 + p^2)^{1/2}$

$$\frac{\mathrm{d}r}{\mathrm{d}t} = \frac{h}{2}r,$$

so that, over a time interval of length h, r is multiplied by the factor $\exp(h^2/2) = 1 + h^2/2 + O(h^3)$. This is precisely what we found in (3.24) for the Euler solution of the harmonic oscillator (but the $O(h^3)$ remainder here will not coincide with the one in (3.24), because there is an $O(h^3)$ difference between ψ_h and $\tilde{\varphi}_h$).

Example 4.5. In (4.1), $\tilde{f}_h(x)$ is a first-degree polynomial in h. If $\tilde{f}_h(x)$ is chosen to be quadratic in h, that is, $\tilde{f}_h = f - (h/2)f'f + h^2 f_2$, it is possible to determine f_2 so as to achieve $\psi_h - \tilde{\varphi}_h = O(h^4)$ for the Euler map $\psi_h(x) = x + hf(x)$. Similarly, taking $\tilde{f}_h(x)$ as a suitable chosen polynomial in h with degree $\mu = 3, 4, \ldots$, it is possible to achieve $\psi_h - \tilde{\varphi}_h = O(h^{\mu+2})$.

4.3.2. Definition
Given a (consistent) integrator ψ_h for system (2.1), there exists a (unique) *formal series in powers of* h,

$$\tilde{f}_h^{\infty}(x) = f(x) + hg^{[1]}(x) + h^2 g^{[2]}(x) + \cdots \tag{4.2}$$

(the $g^{[\mu]}$ map the space $\mathbb{R}^D$ into itself), with the property that, for each $\mu = 0, 1, \ldots$, the flow $\tilde{\varphi}_h^{[\mu]}$ of the modified system

$$\frac{\mathrm{d}}{\mathrm{d}t}x = \tilde{f}_h^{[\mu]}(x), \quad \tilde{f}_h^{[\mu]}(x) = f(x) + hg^{[1]}(x) + \cdots + h^{\mu} g^{[\mu]}(x), \tag{4.3}$$

satisfies

$$\psi_h(x) - \tilde{\varphi}_h^{[\mu]}(x) = O(h^{\mu+2}). \tag{4.4}$$

Furthermore, for a symmetric method, the odd-numbered $g^{[\mu]}(x)$ are identically zero:

$$g^{[1]}(x) = g^{[3]}(x) = g^{[5]}(x) = \cdots = 0. \tag{4.5}$$

As μ increases, the solutions of the modified system (4.3) provide better and better approximations to ψ_h. For some vector fields f and some integrators (see an example in Proposition 4.9 below), it may happen that the formal series (4.2) converges for each x (so that $\tilde{f}_h^{\infty}$ is a well-defined mapping $\mathbb{R}^D :\to \mathbb{R}^D$) and that furthermore the flow $\tilde{\varphi}_h^{[\infty]}$ of

$$\frac{\mathrm{d}}{\mathrm{d}t}x = \tilde{f}_h^{\infty}(x) \tag{4.6}$$

coincides with ψ_h. In those cases, (4.6) provides an *exact* modified system to study ψ_h. However, such situations are exceptional, because, in general, discrete dynamical systems (such as the one generated by ψ_h) possess features that cannot appear in flows of autonomous differential systems. It is the lack of convergence of (4.2) that makes it necessary to consider the truncations in (4.3) which do not exactly reproduce ψ_h but approximate it with an error as in (4.4). It is possible, by increasing μ as h becomes smaller and smaller, to render the discrepancy $\psi_h(x) - \tilde{\varphi}_h^{[\mu(h)]}(x)$ exponentially small with respect to h, as first proved by Neishtadt (Sanz-Serna and Calvo 1994, Section 10.1).

4.3.3. Finding explicitly the modified equations

For splitting integrators, the terms of the series (4.2) may be found explicitly by using the Baker–Campbell–Hausdorff formula. The next theorem provides a summary of some key points; a more detailed description is given in Section 9.1.

Theorem 4.6. Assume that system (3.7) is integrated by means of a (consistent) splitting algorithm of the general format (3.21). The series (4.2) is of the form

$$(f^{(\mathrm{A})} + f^{(\mathrm{B})}) + hC_{1,1}[f^{(\mathrm{A})}, f^{(\mathrm{B})}] + h^2(C_{2,1}[f^{(\mathrm{A})}, [f^{(\mathrm{A})}, f^{(\mathrm{B})}]] + C_{2,2}[f^{(\mathrm{B})}, [f^{(\mathrm{A})}, f^{(\mathrm{B})}]]) + \cdots$$

where, for each $\mu = 1, 2 \ldots$, the coefficient of h^μ is a linear combination of linearly independent iterated commutators involving $\mu + 1$ fields $f^{(\mathrm{A})}$, $f^{(\mathrm{B})}$. The coefficients $C_{i,j}$ that appear in the linear combinations are known polynomials on the coefficients a_k and b_ℓ that appear in the formula (3.21).

If the splitting is palindromic then all the coefficients $C_{2i+1,j}$ corresponding to odd powers of h vanish.

For Runge–Kutta methods it is also possible to give expressions for the terms in the series (4.2): see Calvo, Murua and Sanz-Serna (1994) or Sanz-Serna (1996).

4.3.4. Modified equations and order

If in (4.2) the functions $g^{[1]}, \ldots, g^{[\mu-1]}, \mu \geq 2$, vanish so that

$$\tilde{f}_h^{[\mu]}(x) - f(x) = h^\mu g^{[\mu]}(x) + O(h^{\mu+1}),$$

then the flow $\tilde{\varphi}_h^{[\mu]}$ of (4.3) and the flow φ_h of (2.1) will satisfy

$$\tilde{\varphi}_h^{[\mu]}(x) - \varphi_h(x) = h^{\mu+1} g^{[\mu]}(x) + O(h^{\mu+2}),$$

which in view of (4.4) implies that

$$\psi_h(x) - \varphi_h(x) = h^{\mu+1} g^{[\mu]}(x) + O(h^{\mu+2}),$$

that is, the integrator $\psi_h(x)$ has order μ (or higher). The converse is also true, because the argument may be reversed: order $\geq \mu$ implies that $g^{[1]}, \ldots, g^{[\mu-1]}$ must vanish. From (4.5), the order of a symmetric method must be an even integer. To sum up, we have the following result, which makes it possible to use the series (4.2) rather than the mapping ψ_h to study the order of a given integrator.

Theorem 4.7. A (consistent) integrator has order $\geq \nu$, $\nu \geq 2$, if and only if the functions $g^{[1]}, \ldots, g^{[\nu-1]}$ appearing in (4.2) are identically zero.

When the order is exactly ν, $\nu = 1, 2, \ldots$, the leading term of the truncation error is $h^{\nu+1} g^{[\nu]}(x)$.

A (consistent) symmetric integrator has even order.

This theorem, in tandem with Theorem 4.6, provides the standard way to write down the order conditions for splitting methods, that is, the relations on the coefficients a_k, b_ℓ that are necessary and sufficient for an integrator to have order ν. Specifically, order $\geq \nu$ is equivalent to $C_{i,j} = 0$ whenever $i < \nu$. Here are the simplest illustrations.

- The Lie–Trotter formula (3.11) has $\nu = 1$, and, according to Theorems 4.6 and 4.7, the leading term of the local error is a constant multiple of $h^2[f^{(\mathrm{A})}, f^{(\mathrm{B})}]$, which matches our earlier finding in (3.12).
- The condition $C_{1,1} = 0$ is necessary and sufficient for an integrator to have order $\nu \geq 2$. It is automatically satisfied for palindromic formulas.
- Strang's formula (3.13) has order exactly 2. The coefficient of h^3 in the expansion of the local error is a combination of iterated commutators with three fields. This was already found in (3.14).
- To have order $\nu \geq 3$ we have to impose the order conditions $C_{1,1} = 0$, $C_{2,1} = 0$, $C_{2,2} = 0$.

4.3.5. Modified equations and geometric integrators

The geometric properties of integrators have a clear impact on their modified systems.

Let us begin with the Hamiltonian case. If the split fields in (3.8) are Hamiltonian with Hamiltonian functions $H^{(\mathrm{A})}$ and $H^{(\mathrm{B})}$, then by invoking Theorem 2.13 it is feasible to work with iterated Poisson brackets of $H^{(\mathrm{A})}$ and $H^{(\mathrm{B})}$ rather than with the iterated Lie brackets of $f^{(\mathrm{A})}$ and $f^{(\mathrm{B})}$. Then the expansion in Theorem 4.6 is replaced by

$$(H^{(\mathrm{A})} + H^{(\mathrm{B})}) - hC_{1,1}\{H^{(\mathrm{A})}, H^{(\mathrm{B})}\} + h^2(C_{2,1}\{H^{(\mathrm{A})}, \{H^{(\mathrm{A})}, H^{(\mathrm{B})}\}\} + C_{2,2}\{H^{(\mathrm{B})}, \{H^{(\mathrm{A})}, H^{(\mathrm{B})}\}\}) + \cdots .$$

For each μ, *the modified system* (4.3) *is a Hamiltonian system.* This is a reflection of the fact that, according to Theorem 4.1, splitting integrators

give rise to mappings ψ_h that are symplectic. In the class of Runge–Kutta methods it is also true that symplectic integrators have modified systems that are Hamiltonian.

Speaking informally, we may say that all integrators change the system being integrated into a modified system; in non-symplectic methods the (perhaps small) change is such that the modified system is no longer Hamiltonian. Symplectic methods are those that change Hamiltonian systems into Hamiltonian systems. This heuristic description cannot be made entirely rigorous because, as pointed out above, the exact modified system (4.6) only exists in a formal sense due to the lack of convergence of the series in (4.2). The existence of a Hamiltonian modified system is at the basis of many favourable properties of symplectic integrators.

Remark 4.8. The considerations above only make sense if the step size h is held constant along the integration interval. Since the modified system changes with h, variable step size implementations of symplectic integrators do not have well-defined modified systems and in fact their behaviour is closer to that of non-symplectic integrators than to that of symplectic integrators used with constant step sizes (Calvo and Sanz-Serna 1993).

For the reversible case, consider the situation of Theorem 4.2. From Proposition 2.14 all iterated commutators involving an odd number of fields are themselves S-reversible. Since, according to Theorem 4.6, for *palindromic splitting integrators* the iterated commutators with an even number of fields enter the expansion (4.2) with null coefficients, then *the modified systems* (4.3) *are reversible*.

4.3.6. Conservation of energy by symplectic integrators

As noted above, Neishtadt proved that, under suitable regularity assumptions, the modified system may be chosen so as to ensure that its h-flow is exponentially close to the mapping ψ_h. For symplectic integrators, the modified system is Hamiltonian and therefore exactly preserves its own Hamiltonian function that we denote by $\tilde{H}_h$. It follows that, except for exponentially small errors, ψ_h preserves $\tilde{H}_h$. On the other hand, for a symplectic integrator of order ν, the difference between $\tilde{H}_h$ and H is $O(h^\nu)$ (Theorem 4.7). These considerations make it possible to prove that symplectic integrators preserve the value of the Hamiltonian H of the system being integrated with error $O(h^\nu)$ over time intervals $0 \leq t \leq T_h$ whose length T_h increases exponentially as $h \to 0$ (Hairer *et al.* 2010).

For linear problems an exact modified system exists, and using the same argument, we may conclude that the error in energy of a symplectic integrator has an $O(h^\nu)$ bound over the infinite interval $0 \leq t < \infty$, or in other words the energy error may be bounded independently of the number of

step taken. For the case of the harmonic oscillator, this will be illustrated presently.

4.4. Geometric integrators and the harmonic model problem

We take again the integration of the harmonic oscillator as a model problem (see (3.22)–(3.23)); now our interest is in studying in detail the behaviour of geometric integrators.

We focus on (consistent) integrators that are both *symplectic and reversible.* In terms of the matrix $\tilde{M}_h$, the first of these properties corresponds to $A_h D_h - B_h C_h = 1$ and, when this condition holds, reversibility is equivalent to $A_h = D_h$. Our treatment follows that of Blanes *et al.* (2014).

The characteristic polynomial of $\tilde{M}_h$ is of the form (3.25) and there are four possibilities, the first two corresponding to *unstable* simulations and the last two corresponding to *stable* simulations.

(1) h is such that $|A_h| > 1$. In that case $\tilde{M}_h$ has spectral radius > 1 and therefore the powers $\tilde{M}_h^n$ grow exponentially with n.

(2) $A_h = \pm 1$ and $|B_h| + |C_h| > 0$. The powers $\tilde{M}_h^n$ grow linearly with n.

(3) $A_h = \pm 1$, $B_h = C_h = 0$, that is, $\tilde{M}_h = \pm I$, $\tilde{M}_h^n = (\pm I)^n$.

(4) h is such that $|A_h| < 1$. In that case, $\tilde{M}_h$ has complex conjugate eigenvalues of unit modulus and the powers $\tilde{M}_h^n$, $n = 0, 1, \ldots$, remain bounded.

Comparing (3.22) with the result of setting $t = h$ in (2.4), we see that, by consistency, $B_h = h + O(h^2)$ and $C_h = -h + O(h^2)$, and therefore $A_h = (1 + B_h C_h)^{1/2} = 1 - h^2/2 + O(h^3)$. Thus, for $h > 0$ sufficiently small, $A_h < 1$ and the integration will be stable. The *stability interval* of the integrator is the longest interval $(0, h_{\max})$ such that integrations with $h \in (0, h_{\max})$ are stable. For reasons discussed in Example 3.14, methods with long stability intervals are often to be favoured. From Example 3.7, the midpoint rule has stability interval $(0, \infty)$. Explicit integrators have stability intervals of finite length.

For each h such that $|A_h| \leq 1$, it is expedient to introduce $\theta_h \in \mathbb{R}$ such that $A_h = D_h = \cos\theta_h$. For $|A_h| < 1$, we have $\sin\theta_h \neq 0$ and we may define

$$\chi_h = B_h / \sin\theta_h. \tag{4.7}$$

In terms of θ_h and χ_h, the matrices in (3.22) and (3.23) are then

$$\tilde{M}_h = \begin{bmatrix} \cos\theta_h & \chi_h \sin\theta_h \\ -\chi_h^{-1} \sin\theta_h & \cos\theta_h \end{bmatrix} \tag{4.8}$$

and

$$\tilde{M}_h^n = \begin{bmatrix} \cos(n\theta_h) & \chi_h \sin(n\theta_h) \\ -\chi_h^{-1} \sin(n\theta_h) & \cos(n\theta_h) \end{bmatrix}. \tag{4.9}$$

For a value of h in the (stable) case $A_h = \pm 1$, $B_h = C_h = 0$, we have $\sin\theta_h = 0$, so (4.7) does not make sense. However, the matrix $\tilde{M}_h$ is of the form (4.8) for any choice of χ_h. (Typically, for such a value of h, one may avoid the indeterminacy in the value of χ_h by taking limits as $\epsilon \to 0$ in $\chi_{h+\epsilon} = B_{h+\epsilon}/\sin\theta_{h+\epsilon}$.)

For a method of order ν, $\chi_h = 1 + O(h^\nu)$, $\theta_h = h + O(h^{\nu+1})$ as $h \to 0$. By comparing the numerical $\tilde{M}_h^n$ in (4.9) with the true M_{nh} in (2.4), one sees that a method with $\theta_h = h$ would have no phase error: the angular frequency of the rotation of the numerical solution would coincide with the true angular frequency of the harmonic oscillator. More generally, the difference $\theta_h - h$ governs the phase error. According to (4.9), this phase error grows *linearly* with n (recall Table 3.1). On the other hand, a method with $\chi_h = 1$ would have no energy error: the numerical solution would remain on the correct level curve of the Hamiltonian, *i.e.* on the circle $p^2+q^2 = p_0^2+q_0^2$. The discrepancy between χ_h and 1 governs the energy errors. In (4.9) we see that these are *bounded* as n grows.

The preceding considerations may alternatively be understood by considering the modified Hamiltonian given in the next result.

Proposition 4.9. Consider the application to the harmonic oscillator (2.3) of a (consistent) reversible volume-preserving integrator (3.22), and assume that the step size h is stable, so that $\tilde{M}_h$ may be written in the form (4.8). Then ψ_h exactly coincides with the h flow of the modified Hamiltonian

$$\tilde{H}_h = \frac{\theta_h}{2h}\left(\chi_h p^2 + \frac{1}{\chi_h} q^2\right).$$

In particular, numerical trajectories are contained in ellipses,

$$\chi_h p^2 + \frac{1}{\chi_h} q^2 = \chi_h p_0^2 + \frac{1}{\chi_h} q_0^2, \tag{4.10}$$

rather than in circles (Figure 4.1).

Remark 4.10. A comparison of a given integrator (3.22) with (2.4) shows that

$$\begin{bmatrix} q_{n+1} \\ p_{n+1} \end{bmatrix} = \tilde{M}_h \begin{bmatrix} q_n \\ p_n \end{bmatrix}, \quad \tilde{M}_h = \begin{bmatrix} A_h & -C_h \\ -B_h & D_h \end{bmatrix} \tag{4.11}$$

is a second integrator of the same order of accuracy. We might think that (4.11) arises from (3.22) by changing the roles of the variables q and p. In the particular case of splitting integrators, (4.11) arises from (3.22) after swapping the roles of the split systems A and B. The integrators (3.22) and (4.11) share the same interval of stability and the same θ_h. The function χ_h of (4.11) is obtained by changing the sign of the reciprocal of the function χ_h of (3.22). The important function $\rho(h)$ to be introduced in Proposition 6.4 is also the same for (3.22) and (4.11). The velocity Verlet algorithm and

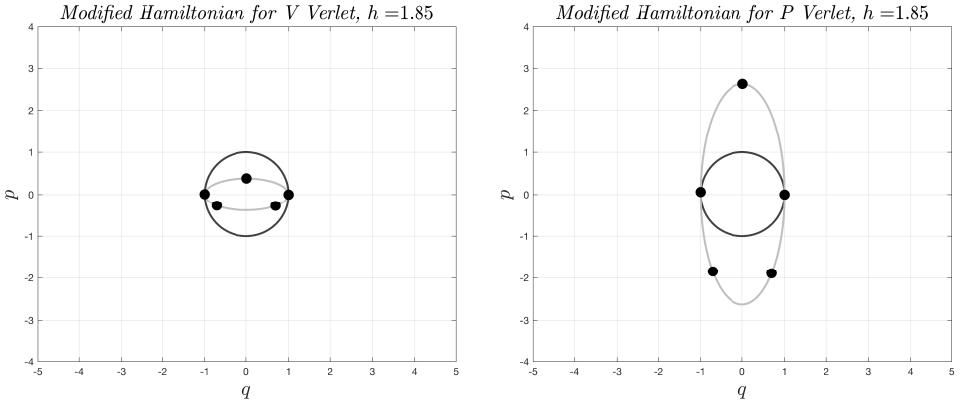


Figure 4.1. Modified Hamiltonian for (a) velocity and (b) position Verlet. The dots are the first five points along a discrete orbit of the velocity/position Verlet integrator initiated at $(q, p) = (1, 0)$ and the grey line provides the associated level set of the modified Hamiltonian. For comparison, the black contour line shows the corresponding level set of the true Hamiltonian.

the position Verlet algorithm provide an example of this kind of pair of integrators (see Examples 3.8 and 3.9).

The results we have just presented may be extended to quadratic Hamiltonians with d degrees of freedom (3.27): it is sufficient to use diagonalization as in Proposition 3.13. In particular, for stability we require that the stability interval of the integrator contains all products $h\omega_i$, where the frequencies ω_i are the square roots of the eigenvalues of $L^{-1}KL^{-T}$, with $LL^T = M$.

4.5. Optimal stability of Strang's method

Let us fix an integer N and consider consistent palindromic splitting integrators (3.19)–(3.20) with $s = N$; these use N evaluations of F per step when applied to problems of the form (3.9)–(3.10). The corresponding coefficient A_h in (3.22) is a polynomial of degree N in the variable $z = h^2$ (for obvious reasons A_h is often called the *stability polynomial* of the integrator). We pointed out above that consistency imposes the relation $A_h = 1 - z/2 + O(h^2)$. Our aim is to identify, among the class just described, the polynomial $A_h(z)$ that satisfies $|A_h(z)| < 1$ for $0 < h < h_{\max}$ with $h_{\max}$ as large as possible. As we shall show presently, the $A_h(z)$ sought corresponds to the integrator

$$\psi_h = \overbrace{\psi^V_{h/N} \circ \cdots \circ \psi^V_{h/N}}^{N \text{ times}}, \tag{4.12}$$

where ψ_h^V is the mapping associated with the Strang/Verlet formula (3.13).[7] Note that to carry out a step of length h with the method in (4.12), one just has to take N consecutive steps of length h/N of standard velocity Verlet. In other words, *subject to stability, if one wishes to take as long a step as possible with a budget of N evaluations of the force per step, the best choice is to concatenate N steps of Strang/Verlet.*[8]

To see the optimality of ψ_h in (4.12), we first note that, after expressing $\psi_{h/N}^V$ in terms of the A and B flows and merging consecutive B flows, the mapping (4.12) corresponds indeed to a palindromic splitting with N stages. Then, from Example 3.8 we know that Verlet is stable for $0 < h < 2$ and this implies that (4.12) is stable for $0 < h/N < 2$ (for (4.12) the powers of $\tilde{M}_h$ are powers of the Verlet matrix $\tilde{M}_{h/N}^V$). In this way (4.12) has stability interval $(0, 2N)$ and we shall prove next that this is the longest possible. From Example (3.8), Verlet with step size h/N has $A_{h/N}^V = 1 - h^2/(2N^2)$, which, in view of (4.8)–(4.9), implies that for (4.12) the coefficient A_h has the expression

$$\cos\theta_h = \cos(N\theta_{h/N}^V).$$

Recalling the definition $T_N(\cos\alpha) = \cos(N\alpha)$ of the Chebyshev polynomial T_N, we observe that for (4.12)

$$A_h(z) = T_N\left(1 - \frac{z}{2N^2}\right).$$

Well-known properties of T_N imply that no other polynomial $A_h(z)$ of degree $\leq N$ with $A_h = 1 - z/2 + O(h^2)$ has modulus ≤ 1 in the interval $-1 < 1 - z/(2N^2) < 1$, that is, when $0 < h < 2N$.

5. Monte Carlo methods

In this section we review some basic concepts and principles of Monte Carlo methods aimed at computing integrals with respect to a given probability distribution (Sokal 1997, Asmussen and Glynn 2007, Liu 2008, Diaconis 2009). We also describe the Hamiltonian or hybrid Monte Carlo (HMC) method and some of its variants.

5.1. Simple Monte Carlo methods

Given a probability distribution μ in $\mathbb{R}^D$ (the *target distribution*) and a function $F : \mathbb{R}^D \to \mathbb{R}$, the problem addressed by the algorithms considered

[7] Of course, if, instead of integrators of the format (3.19)–(3.20), one is interested in the corresponding palindromic integrators that start and end with an A flow, then, on the right-hand side of (4.12), one has to use (3.15) rather than (3.13).

[8] More precisely, if one is interested in methods that start with a kick (resp. drift) one has to concatenate the velocity (resp. position) version of Verlet.

here is to numerically estimate the following D-dimensional integral with respect to μ:

$$\mu(F) = \int_{\mathbb{R}^D} F(x)\mu(\mathrm{d}x). \tag{5.1}$$

In general, $\mu(F)$ cannot be determined analytically. Moreover, since the dimension D might not be small, conventional numerical quadrature is likely not to be practical or even feasible.

The simple Monte Carlo method approximately computes $\mu(F)$ in (5.1) by generating N independent and identically distributed (i.i.d.) samples $X_1, \ldots, X_N$ from μ, evaluating the function F at these samples and using the *estimator*

$$\bar{F}_N = \frac{1}{N}\sum_{i=1}^{N} F(X_i). \tag{5.2}$$

Assuming that $\mu(F) < \infty$, the *law of large numbers* states that

$$\lim_{N\to\infty} \bar{F}_N = \mu(F) \quad \text{as } N \to \infty,$$

almost surely. If, in addition, the standard deviation $\sigma_0(F)$ of the random variable $F(X)$, $X \sim \mu$, defined by

$$\sigma_0(F)^2 = \int_{\mathbb{R}^D} (F(x) - \mu(F))^2 \mu(\mathrm{d}x) \tag{5.3}$$

is finite, the *central limit theorem* ensures the following distributional limit:

$$N^{1/2}(\bar{F}_N - \mu(F)) \xrightarrow{d} \mathcal{N}(0, \sigma_0(F)^2) \quad \text{as } N \to \infty.$$

Loosely speaking, this may be interpreted as stating that the distribution of $\bar{F}_N$ is approximately $\mathcal{N}(0, \sigma_0(F)^2/N)$. Hence, the standard deviation of the error $\bar{F}_N - \mu(F)$ decreases like the inverse square root of the number of samples. Often this standard deviation is referred to as *Monte Carlo error*. Thus, to halve the Monte Carlo error the number of i.i.d. samples needs to be quadrupled.

In most cases of practical interest, we cannot directly generate i.i.d. samples X_i from μ and must therefore resort to Markov chain Monte Carlo methods.

5.2. Markov chain Monte Carlo methods

Recall that a *Markov chain* with state space $\mathbb{R}^D$ is a sequence of random D-vectors $\{X_i\}_{i\in\mathbb{N}}$ that satisfies the Markov property

$$\mathbb{P}(X_{i+1} \in A \mid X_1, \ldots, X_i) = \mathbb{P}(X_{i+1} \in A \mid X_i)$$

for all measurable sets A. In other words, given the past history of the chain $X_1, \ldots, X_i$, the only information required to update the state of the chain is

the current state X_i. Here our interest is restricted to (time-) homogeneous chains, that is, to cases where $\mathbb{P}(X_{i+1} \in A \mid X_i)$ is independent of i.

Typically one constructs a homogeneous Markov chain in terms of its *transition probabilities* Π_x, $x \in \mathbb{R}^D$. These are the probabilities

$$\Pi_x(A) = \mathbb{P}(X_{i+1} \in A \mid X_i = x),$$

with $i \in \mathbb{N}$ and A measurable. Often Π_x may be computed as

$$\Pi_x(A) = \int_A \Pi_x(\mathrm{d}x'),$$

for a suitable kernel $\Pi_x(\mathrm{d}x')$. Clearly the chain is determined once the transition probabilities and the distribution of X_1 are known. In practice, the term 'chain' is used in a wide sense to refer to the transition probabilities without specifying the distribution of X_1.

A probability distribution ν is an *invariant or stationary distribution* of a Markov chain with transition probabilities Π_x if

$$\int_{\mathbb{R}^D} \Pi_x(A)\nu(\mathrm{d}x) = \int_A \nu(\mathrm{d}x)$$

holds for all measurable sets A. We also say that Π_x preserves ν. In the situations we are interested in, a Markov chain will have a unique invariant distribution. If ν is the invariant distribution of the chain and in addition $X_1 \sim \nu$, one says that the chain is *at stationarity*.

Markov chain Monte Carlo (MCMC) methods generate a Markov chain $\{X_i\}_{i\in\mathbb{N}}$ that has the target μ as an invariant distribution and estimate $\mu(F)$ by the average (5.2). By analogy to the simple i.i.d. situation described above, one would like to have MCMC methods that meet two basic requirements.

- For each $F : \mathbb{R}^D \to \mathbb{R}$ such that $\mu(F) < \infty$,

$$\bar{F}_N \overset{\text{a.s.}}{\to} \mu(F) \quad \text{as } N \to \infty. \tag{5.4}$$

  This is the MCMC analogue of the law of large numbers.

- For each $F : \mathbb{R}^D \to \mathbb{R}$ such that $\mu(F) < \infty$ and $\sigma_0(F) < \infty$,

$$N^{1/2}(\bar{F}_N - \mu(F)) \overset{d}{\to} \mathcal{N}(0, \sigma(F)^2) \quad \text{as } N \to \infty, \tag{5.5}$$

  for some $\sigma(F)$. This is the MCMC analogue of the central limit theorem.

For each fixed function F and Markov chain $\{X_i\}_{i\in\mathbb{N}}$, the constant $\sigma(F)^2$ appearing in (5.5) is called the *asymptotic variance* of the MCMC estimator $\bar{F}_N$. A straightforward calculation shows that this asymptotic variance satisfies

$$\sigma(F)^2 = \sigma_0(F)^2 + 2\sum_{i>1} \mathrm{cov}_\mu(F(X_1), F(X_i)),$$

where $\sigma_0(F)$ is defined in (5.3) and the covariances are computed assuming that the chain is at stationarity. If the X_i were independent, all the covariances $\mathrm{cov}_\mu(F(X_1), F(X_i))$ would vanish and we would recover the standard central limit theorem. Since, in most interesting cases, the X_i in the Markov chain are not mutually independent, $\sigma(F)$ is often larger than $\sigma_0(F)$. Generally speaking it is desirable to have low values of $\mathrm{cov}_\mu(X_1, X_i)$ so that $\sigma(F)$ is not far away from $\sigma_0(F)$ for each F.

In practice, the inputs that the user has to supply to an MCMC algorithm include the following.

- A sample of the initial state X_1. Ideally, this sample should be taken in a domain of state space of high probability. Otherwise the chain may need many steps to start generating useful samples. A discussion of this issue is beyond the scope of this paper.
- A possibly non-normalized density function ρ of the target μ, that is, the probability density function is $Z^{-1}\rho(x)$, where $Z = \int_{\mathbb{R}^D} \rho(x)\,\mathrm{d}x$ is not assumed to be 1; the value of Z is not required to run the algorithms.

5.3. Metropolis method for reversible maps

The replacement of the i.i.d. variables that simple Monte Carlo uses in the estimator (5.2) with variables of a Markov chain is of interest because it is not difficult to construct a chain that has a given target μ as an invariant distribution. The key to this construction is the Metropolis–Hastings accept/reject mechanism, which turns a given *proposal* chain (for which μ is not invariant) into a *Metropolized chain*, which leaves μ invariant. The simplest Metropolis rule was introduced in 1953 in a landmark paper (Metropolis *et al.* 1953); later, Hastings provided an important generalization (Hastings 1970).

A review of the Metropolis–Hastings rule is not required for our purposes. However, we shall present a trimmed-down variant of Metropolis–Hastings that we will use to define HMC. This variant works in the special case where the target μ is invariant with respect to a linear involution, takes as input a reversible deterministic map, and manufactures a Markov chain that preserves μ. While the chain that we construct is not expected to satisfy a law of large numbers and therefore has no practical merit, Proposition 5.2 will be used later to analyse HMC methods.

The technique, patterned after Fang, Sanz-Serna and Skeel (2014), requires a non-normalized density function $\rho : \mathbb{R}^D \to \mathbb{R}$ of the target, and, as pointed out above, a map $\Phi : \mathbb{R}^D \to \mathbb{R}^D$ that is reversible with respect to a linear involution S that preserves probability, that is, $\rho \circ S = \rho$. The accept/reject mechanism is based on the *acceptance probability* $\alpha : \mathbb{R}^D \to [0, 1]$

defined as

$$\alpha(x) = \min\left\{1, \frac{\rho(\Phi(x))}{\rho(x)}|\det \Phi'(x)|\right\} \tag{5.6}$$

(Φ' is the Jacobian matrix of Φ).

We consider the following algorithm.

Algorithm 5.1 (Metropolized reversible map). Given $X_0 \in \mathbb{R}^D$ (the input state), the method outputs a state X_1 as follows.

Step 1. Generate a proposal move $\tilde{X}_1 = \Phi(X_0)$.

Step 2. Output $X_1 = \gamma\tilde{X}_1 + (1-\gamma)S(X_0)$, where γ is a Bernoulli random variable with parameter $\alpha(X_0)$ (*i.e.* γ is 1 with probability $\alpha(X_0)$ and 0 with probability $1 - \alpha(X_0)$).

Step 2 contains the accept/reject mechanism. In the case of *acceptance* the updated state coincides with the state proposed $\tilde{X}_1$ from Step 1; in the case of *rejection* the updated state is $S(X_0)$. Note that for rejection, conventional Metropolis mechanisms set the updated state of the chain to be X_0.

Proposition 5.2. In the situation described above, let $\{X_i\}_{i\in\mathbb{N}}$ be the Markov chain defined by iterating Algorithm 5.1. Then the target distribution is an invariant distribution of this chain.

Proof. The transition kernel of $\{X_i\}_{i\in\mathbb{N}}$ is given by

$$\Pi_x(\mathrm{d}x') = \alpha(x)\delta(x' - \Phi(x))\,\mathrm{d}x' + (1-\alpha(x))\delta(x' - S(x))\,\mathrm{d}x',$$

where $\delta(\cdot)$ is the Dirac-delta function. Hence, for any measurable set A, if $1_A(x)$ denotes the corresponding indicator function,

$$\begin{aligned}\int_{\mathbb{R}^D} \Pi_x(A)\rho(x)\,\mathrm{d}x &= \int_A \rho(x)\,\mathrm{d}x \\ &\quad + \int_{\mathbb{R}^D} \alpha(x)\rho(x)1_A(\Phi(x))\,\mathrm{d}x - \int_{\mathbb{R}^D} \alpha(x)\rho(x)1_A(S(x))\,\mathrm{d}x.\end{aligned}$$

By change of variables in the third integral,

$$\begin{aligned}\int_{\mathbb{R}^D} \Pi_x(A)\rho(x)\,\mathrm{d}x &= \int_A \rho(x)\,\mathrm{d}x \\ &\quad + \int_{\mathbb{R}^D} \alpha(x)\rho(x)1_A(\Phi(x))\,\mathrm{d}x \\ &\quad - \int_{\mathbb{R}^D} \alpha(S(\Phi(x)))|\det \Phi'(x)|\rho(\Phi(x))1_A(\Phi(x))\,\mathrm{d}x,\end{aligned}$$

where we used the hypothesis that $\rho \circ S = \rho$. The last two terms on the

right-hand side of this equation cancel because

$$\alpha(S(\Phi(x))) = \min\left\{1, \frac{\rho(x)}{\rho(\Phi(x))}|\det \Phi'(x)|^{-1}\right\}$$
$$= \alpha(x)\rho(x)\frac{1}{\rho(\Phi(x))}|\det \Phi'(x)|^{-1},$$

which follows from the reversibility of Φ, the hypothesis $\rho \circ S = \rho$ and Proposition 2.8. □

5.4. The HMC method: basic idea

We consider a target distribution Π in $\mathbb{R}^d$. If $\mathcal{U} : \mathbb{R}^d \to \mathbb{R}$ denotes the negative logarithm of the (not necessarily normalized) probability density function of the target, then

$$\Pi(\mathrm{d}q) = Z_q^{-1}\exp(-\mathcal{U}(q))\,\mathrm{d}q, \quad \text{where } Z_q = \int_{\mathbb{R}^d}\exp(-\mathcal{U}(q))\,\mathrm{d}q.$$

The Monte Carlo algorithms studied here use $\mathcal{U}$ but do not require knowledge of the normalization factor Z_q. In HMC, regardless of the application in mind, $\mathcal{U}$ is seen as the potential energy of a mechanical system with coordinates q. Then auxiliary momenta $p \in \mathbb{R}^d$ and a quadratic kinetic energy function $\mathcal{T}(p) = (1/2)p^T M^{-1} p$ are introduced as in (2.10) (M is a positive-definite symmetric matrix chosen by the user).[9] The total energy of this fictitious mechanical system is $H = \mathcal{T} + \mathcal{U}$ and the equations of motion are given in (2.9).

Example 5.3. In the particular case where the target μ is Gaussian with non-normalized density $\exp(-(1/2)q^T K q)$, $q \in \mathbb{R}^d$, we have

$$\mathcal{U}(q) = (1/2)q^T K q,$$

which leads to the Hamiltonian (3.27) discussed above. For a univariate standard normal target, $H = (1/2)(p^2+q^2)$ and Hamilton's equations reduce to those of harmonic oscillator (2.3).

The *Boltzmann–Gibbs* distribution in $\mathbb{R}^{2d}$ corresponding to H was discussed earlier in connection with Theorem 2.7. This distribution is defined as

$$\Pi_{BG}(\mathrm{d}q, \mathrm{d}p) = (2\pi)^{-d/2}|\det M|^{-1/2}\exp\left(-\frac{1}{2}p^T M^{-1} p\right)$$
$$\times Z_q^{-1}\exp(-\mathcal{U}(q))\,\mathrm{d}q\,\mathrm{d}p \tag{5.7}$$

[9] Often M is just taken to be the unit matrix. However, M may be advantageously chosen to precondition the dynamics: see Remark 8.4.

(for simplicity the inverse temperature is taken here to be $\beta = 1$). Clearly the target Π is the q-marginal of Π_{BG}. The p-marginal is Gaussian with zero mean and covariance matrix M; therefore samples from this marginal are easily available (and will be put to use in the algorithms below). A key fact for our purposes: Hamilton's equations of motion (2.9) preserve Π_{BG} (Theorem 2.7).

HMC generates (correlated) samples $(q_i, p_i) \in \mathbb{R}^{2d}$ by means of a Markov chain that leaves Π_{BG} invariant; the corresponding marginal $q_i \in \mathbb{R}^d$ chain then leaves invariant the target distribution Π. The basic idea of HMC is encapsulated in the following algorithm (the duration $\lambda > 0$ is a – deterministic – parameter, whose value is specified by the user).

Algorithm 5.4 (exact HMC). Let $\lambda > 0$ denote the duration parameter.

Given the current state of the chain $(q_0, p_0) \in \mathbb{R}^{2d}$, the method outputs a state $(q_1, p_1) \in \mathbb{R}^{2d}$ as follows.

Step 1. Generate a d-dimensional random vector $\xi_0 \sim \mathcal{N}(0, M)$.

Step 2. Evolve over the time interval $[0, \lambda]$ Hamilton's equations (2.9) with initial condition $(q(0), p(0)) = (q_0, \xi_0)$.

Step 3. Output $(q_1, p_1) = (q(\lambda), p(\lambda))$.

Note that p_0 plays no role, since the initial condition starts from ξ_0. Step 1 is referred to as *momentum refreshment* or *momentum randomization*.

It is easy to see that this algorithm succeeds in preserving the distribution Π_{BG}.

Theorem 5.5. Consider the Markov chain $\{(q_i, p_i)\}_{i \in \mathbb{N}}$ defined by iterating Algorithm 5.4. The probability distribution Π_{BG} in (5.7) is an invariant distribution of this chain.

Proof. The transformation $(q_0, p_0) \mapsto (q_0, \xi_0)$ obviously preserves the Boltzmann–Gibbs distribution. The same is true for the transformation $(q_0, \xi_0) \mapsto (q_1, p_1)$, as we saw in Theorem 2.7. □

The transition kernel of this chain is given by

$$\Pi_{(q,p)}(\mathrm{d}q', \mathrm{d}p') = \mathbb{E}\{\delta((q', p') - \varphi_\lambda(q, \xi))\}\, \mathrm{d}q'\, \mathrm{d}p',$$

where the expected value is over $\xi \sim \mathcal{N}(0, M)$.

The most appealing feature of the algorithm is that if λ is sufficiently large, we may hope that the Markov transitions $i \to i+1$ produce values q_{i+1} far away from q_i, thus reducing the correlations in the chain and facilitating the exploration of the target distribution.

5.5. Numerical HMC

Algorithm 5.4 cannot be used in practice because in the cases of interest the exact solution flow φ_λ of Hamilton's equations is not available. It is then necessary to resort to numerical approximations to φ_λ, but, as pointed out in Section 4, numerical methods cannot preserve volume in phase space *and* energy and therefore do not preserve exactly the Boltzmann–Gibbs distribution. To correct the bias introduced by the time discretization error, the numerical solution is Metropolized using Algorithm 5.1. However, this requires the numerical integrator to be *reversible*.

Let Ψ_λ denote a numerical approximation of φ_λ (more precisely, if the step size is h and $n = \lfloor \lambda/h \rfloor$ is the number of steps required to integrate up to $t = \lambda$, then $\Psi_\lambda = \psi_h^n$). In order to use Algorithm 5.1 with Ψ_λ playing the role of Φ and the momentum flip involution (2.13) playing the role of S, we first note (Proposition 2.11) that the momentum flip involution preserves H in (2.10) and, as a consequence, it preserves the Boltzmann–Gibbs distribution. In addition Theorem 2.10 ensures that the Hamiltonian flow is reversible with respect to this involution; it then makes sense (Theorem 4.2) to assume that the integrator chosen is such that Ψ_λ is also reversible. The acceptance probability in (5.6) now reads

$$\alpha(q,p) = \min\{1, \mathrm{e}^{-\Delta H(q,p)} |\det \Psi'_\lambda(q,p)|\}, \tag{5.8}$$

where

$$\Delta H(q,p) = H(\Psi_\lambda(q,p)) - H(q,p)$$

is the energy error (recall that if the integrator were exact $H(\Psi_\lambda(q,p))$ would coincide with $H(q,p)$ by conservation of energy, Theorem 2.6).

Algorithm 5.6 (numerical HMC). Let $\lambda > 0$ denote the duration parameter and let Ψ_λ be a reversible numerical approximation to the Hamiltonian flow φ_λ.

Given the current state of the chain $(q_0, p_0) \in \mathbb{R}^{2d}$; the method outputs a state $(q_1, p_1) \in \mathbb{R}^{2d}$ as follows.

Step 1. Generate a d-dimensional random vector $\xi_0 \sim \mathcal{N}(0, M)$.

Step 2. Find $\Psi_\lambda(q_0, \xi_0)$ by evolving Hamilton's equations (2.9) with a reversible integrator over the time interval $[0, \lambda]$ with initial condition (q_0, ξ_0).

Step 3. Output $(q_1, p_1) = \gamma \Psi_\lambda(q_0, \xi_0) + (1 - \gamma)(q_0, -\xi_0)$, where γ is a Bernoulli random variable with parameter $\alpha(q_0, \xi_0)$ with α as in (5.8).

Theorem 5.7. Consider the Markov chain $\{(q_i, p_i)\}_{i \in \mathbb{N}}$ defined by iterating Algorithm 5.6. The probability distribution Π_{BG} in (5.7) is an invariant distribution of this chain.

Proof. As in the preceding theorem, the transformation $(q_0, p_0) \mapsto (q_0, \xi_0)$ preserves the Boltzmann–Gibbs distribution. The same is true for the transformation $(q_0, \xi_0) \mapsto (q_1, p_1)$ according to Proposition 5.2. □

The transition kernel of the chain is given by

$$\Pi_{(q,p)}(\mathrm{d}q', \mathrm{d}p') = \mathbb{E}\{\alpha(q,\xi)\delta((q',p') - \Psi_\lambda(q,\xi))\}\,\mathrm{d}q'\,\mathrm{d}p' + \mathbb{E}\{(1-\alpha(q,\xi))\delta((q',p') - S(q,\xi))\}\,\mathrm{d}q'\,\mathrm{d}p'.$$

In practice, the acceptance probability (5.8) may not be readily available due to the need to compute $\det \Psi'_\lambda(q,p)$. If the numerical approximation Ψ_λ, in addition to being assumed reversible, is also *volume-preserving* (as it would be for splitting integrators according to Theorem 4.1), then the determinant drops from the formula, and then the acceptance probability

$$\alpha(q,p) = \min\{1, \mathrm{e}^{-\Delta H(q,p)}\} \tag{5.9}$$

becomes easily computable. Variants where preservation of volume does not take place are studied by Fang *et al.* (2014).

Remark 5.8. The states of the Markov chain $\{(q_i, p_i)\}_{i\in\mathbb{N}}$ are not to be confused with the intermediate values of q and p that the numerical integrator generates while transitioning the chain from one state of the chain to the next. Those intermediate values were denoted by (q_n, p_n) in the preceding sections and we have preferred not to introduce additional notation to describe the Markov chain.

Remark 5.9. Theorems 5.5 and 5.7 show that Π_{BG} is an invariant distribution for the chains generated by Algorithms 5.4 and 5.6 respectively. However, they do not guarantee that those chains meet the two basic requirements in (5.4) and (5.5), and indeed a simple example will be presented below where the sequence of values of q generated by those algorithms is $q_0, -q_0, q_0, -q_0, \ldots$ so that the requirements are *not* met. A detailed study of the convergence properties of HMC is beyond the scope of this paper and we limit ourselves to some remarks in Section 9.5.

5.6. Exact randomized HMC

The Hamiltonian flow in Step 2 of Algorithm 5.4 is what, in principle, enables HMC to make large moves in state space that reduce correlations in the Markov chain $\{q_i\}_{i\in\mathbb{N}}$. Roughly speaking, one may hope that by increasing the duration λ, q_1 moves away from q_0, thus reducing correlation. However, simple examples show that this outcome is far from assured.

Indeed, for the univariate standard normal target distribution in Example 5.3, the Hamiltonian flow is a rotation in the (q,p)-plane with period 2π. It is easy to see that if q_0 is taken from the target distribution, as λ

increases from 0 to $\pi/2$, the correlation between q_1 and q_0 decreases and for $\lambda = \pi/2$, q_1 and q_0 are independent. However, increasing λ beyond $\pi/2$ will cause an increase in the correlation, and for $\lambda = \pi$, $q_1 = -q_0$ and the chain is not ergodic. For general distributions, it is likely that a small λ will lead to a highly correlated chain, while choosing λ too large may cause the Hamiltonian trajectory to make a U-turn and fold back on itself, thus increasing correlation (Homan and Gelman 2014). Generally speaking, the performance of HMC may be very sensitive to changes in λ, as first noted by Mackenzie (1989). In order to increase the robustness of the algorithm, Mackenzie suggested randomly varying λ from one Markov transition to the next, and for that purpose he used a uniform distribution in the interval $[\lambda_{\min}, \lambda_{\max}]$.

Recently Bou-Rabee and Sanz-Serna (2017) have studied an algorithm where the lengths of the time intervals of integration of the Hamiltonian dynamics at the different transitions of the Markov chain are independent and identically distributed exponential random variables with mean λ; these durations are of course taken to be independent of the state of the chain. The algorithm is then as follows.

Algorithm 5.10 (exact RHMC). Given the current state of the chain $(q_0, p_0) \in \mathbb{R}^{2d}$, the algorithm outputs the state $(q_1, p_1) \in \mathbb{R}^{2d}$ as follows.

Step 1. Generate a d-dimensional random vector $\xi_0 \sim \mathcal{N}(0, M)$.

Step 2. Generate a random duration $t \sim \mathrm{Exp}(1/\lambda)$.

Step 3. Evolve Hamilton's equations (2.2) over the time interval $[0, t]$ with initial condition $(q(0), p(0)) = (q_0, \xi_0)$.

Step 4. Output $(q_1, p_1) = (q(t), p(t))$.

Analogous to Theorem 5.5, the probability distribution Π_{BG} in (5.7) is an invariant distribution of the Markov chain $\{(q_i, p_i)\}_{i\in\mathbb{N}}$ defined by iterating Algorithm 5.10. Analytical results and numerical experiments (Bou-Rabee and Sanz-Serna 2017, §4–5) show that the dependence of the performance of Algorithm 5.10 on the mean duration parameter λ is simpler than the dependence of the performance of Algorithm 5.4 on its constant duration parameter.

5.7. Numerical randomized HMC

Unfortunately, the complex dependence of correlation on the duration parameter λ of Algorithm 5.4 is not removed by time discretization and is therefore inherited by Algorithm 5.6. For instance, for the univariate standard normal target, it is easy to check that if λ is close to an integer multiple of π and $h > 0$ is suitably chosen, then a Verlet numerical integration will result,

for each q_0, in $q_1 = -q_0$ (a move that will be accepted by the Metropolis–Hastings step).

To improve its performance, Algorithm 5.6 is typically operated with values of h that are randomized (Neal 2011). Since, due to stability restrictions, explicit integrators cannot be used with arbitrarily large values of the time-step, h is typically chosen from a uniform distribution in an (often narrow) interval $(\Delta t_{\min}, \Delta t_{\max})$. The number of time-steps N in each integration leg is kept constant, and therefore the length of the integration intervals is random with a uniform distribution in $(N\Delta t_{\min}, N\Delta t_{\max})$. Even after such a randomization, the fact remains that increasing the duration parameter will increase the computational cost of each integration leg and may impair the quality of the sampling.

Several modifications of Algorithm 5.10 that use numerical integration are suggested in Bou-Rabee and Sanz-Serna (2017). The most obvious of them approximates the Hamiltonian flow in Step 3 by a volume-preserving reversible integrator (such as Verlet) operated with a fixed step size h and with the number of integration steps m at the different transitions of the Markov chain being independent and identically distributed geometric random variables with mean λ/h. These random numbers are of course taken to be independent of the state of the chain. As in Algorithm 5.6, one needs an accept/reject mechanism to remove the bias due to the energy errors introduced by this integrator.

Algorithm 5.11 (numerical RHMC). Let $\lambda > 0$ denote the duration parameter and let ψ_h be a reversible numerical approximation of the Hamiltonian flow φ_h.

Given the current state of the chain $(q_0, p_0) \in \mathbb{R}^{2d}$, the algorithm outputs the state $(q_1, p_1) \in \mathbb{R}^{2d}$ as follows.

Step 1. Generate a d-dimensional random vector $\xi_0 \sim \mathcal{N}(0, M)$.

Step 2. Generate a geometric random variable m supported on the set $\{1, 2, 3, \ldots\}$ and with mean λ/h.

Step 3. Output $(q_1, p_1) = \gamma\psi_h^m(q_0, \xi_0) + (1 - \gamma)(q_0, -\xi_0)$, where γ is a Bernoulli random variable with parameter α defined as in (5.9).

Analogous to Theorem 5.7, the probability distribution Π_{BG} in (5.7) is an invariant distribution of the Markov chain $\{(q_i, p_i)\}_{i\in\mathbb{N}}$ defined by iterating Algorithm 5.11.

In the remainder of this paper our attention is focused on Algorithm 5.6. Experiments based on Algorithm 5.11 are reported in Section 8.2 and an example that illustrates the effects of randomization of h and λ is presented in Section 9.3.

6. Numerical integration and HMC

The computational work in HMC stems mainly from the cost of the evaluations of the force $-\nabla\mathcal{U}$ that are required in the numerical integration to be carried out at each transition of the Markov chain (Step 2 of Algorithm 5.6). If the dimension d of q is high, those evaluations are likely to be expensive; for instance, in a molecular dynamics study of a macromolecule with $N \gg 1$ atoms, $d = 3N$ and each atom typically interacts with all others, so that the complexity of evaluating $-\nabla\mathcal{U}$ grows like N^2. As in any other numerical integration, the aim is to reach a target accuracy with the minimum possible complexity. Specific to the HMC scenario is the fact that, in the event of rejection in Step 3, the algorithm wastes all of the force evaluations used to compute $\Psi_\lambda(q,\xi)$. In addition, when a rejection occurs at a transition $i \to i+1$ of the Markov chain, the new value q_{i+1} coincides with the old q_i, and this contributes to an increase in the correlations along the chain, which degrades the quality of sampling as pointed out when discussing the central limit theorem in (5.5). Since low acceptance rates are unwelcome and the acceptance probability (5.9) is a function of the *energy* error $\Delta H(q,p) = H(\Psi_\lambda(q,p)) - H(q,p)$, it is important to perform the integration so as to have *small energy errors*. In this connection we recall from Section 4 that symplectic integrators conserve energy with errors that are $O(h^\nu)$ over exponentially long time intervals. In this way, the symplecticness of the integrator plays a dual role in HMC. On the one hand, it ensures conservation of volume, thereby making it possible to have the simple expression (5.9) for the acceptance probability. On the other hand, it ensures favourable energy errors even if the integration legs are very long.

An additional point: the *sign* of the error matters. Formula (5.9) shows that, if the integration starts from a point (q,p) for which $\Delta < 0$, then $\alpha = 1$, leading to acceptance.

This section begins with a key result, Theorem 6.2, which shows that for reversible volume-preserving integrators the energy error is on average much smaller than one may have anticipated. After that we study in detail the model case of Gaussian targets and discuss the construction of integrators more efficient than Verlet.

6.1. Mean energy error

Step 2 in Algorithm 5.6 requires a volume-preserving reversible integrator if (5.9) is to be used. As we shall discuss now, those geometric properties have a direct impact on the mean energy error. We begin with an auxiliary result (Beskos *et al.* 2013, Lemma 3.3) that holds for any volume-preserving reversible map.

Proposition 6.1. Let $\Psi : \mathbb{R}^{2d} \to \mathbb{R}^{2d}$ be a volume-preserving bijection that is reversible with respect to the momentum flip involution (2.13), and

set $\Delta(q,p) = H(\Psi(q,p)) - H(q,p)$. If g is an odd real function of a real variable, then

$$\int_{\mathbb{R}^{2d}} g(\Delta(q,p))\,\mathrm{e}^{-H(q,p)}\,\mathrm{d}q\,\mathrm{d}p = -\int_{\mathbb{R}^{2d}} g(\Delta(q,p))\,\mathrm{e}^{-H(\Psi(q,p))}\,\mathrm{d}q\,\mathrm{d}p,$$

provided that one of the integrals exists. If g is even, then

$$\int_{\mathbb{R}^{2d}} g(\Delta(q,p))\,\mathrm{e}^{-H(q,p)}\,\mathrm{d}q\,\mathrm{d}p = \int_{\mathbb{R}^{2d}} g(\Delta(q,p))\,\mathrm{e}^{-H(\Psi(q,p))}\,\mathrm{d}q\,\mathrm{d}p,$$

provided that one of the integrals exists.

Proof. By a change of variables under the map $(q,p) \mapsto S(\Psi(q,p))$,

$$\begin{aligned}\int_{\mathbb{R}^{2d}} g(\Delta(q,p))\,\mathrm{e}^{-H(q,p)}\,\mathrm{d}q\,\mathrm{d}p &= \int_{\mathbb{R}^{2d}} g(\Delta(S(\Psi(q,p))))\,\mathrm{e}^{-H(S(\Psi(q,p)))}\,\mathrm{d}q\,\mathrm{d}p \\ &= \int_{\mathbb{R}^{2d}} g(-\Delta(q,p))\,\mathrm{e}^{-H(\Psi(q,p))}\,\mathrm{d}q\,\mathrm{d}p,\end{aligned}$$

where in the first step we used that both S and Ψ preserve volume and in the second we took into account that S leaves H invariant and that the reversibility of Ψ implies

$$\Delta(q,p) = -\Delta(S(\Psi(q,p))). \tag{6.1}$$

□

We now use the proposition to bound the average of Δ.

Theorem 6.2. Let $\Psi : \mathbb{R}^{2d} \to \mathbb{R}^{2d}$ be a bijection that is volume-preserving and reversible with respect to the momentum flip involution (2.13). If the integral

$$m_\Delta = \int_{\mathbb{R}^{2d}} \Delta(q,p)\,\mathrm{e}^{-H(q,p)}\,\mathrm{d}q\,\mathrm{d}p, \tag{6.2}$$

with $\Delta(q,p) = H(\Psi(q,p)) - H(q,p)$, exists, then

$$0 \le m_\Delta \le \int_{\mathbb{R}^{2d}} \Delta(q,p)^2\,\mathrm{e}^{-H(q,p)}\,\mathrm{d}q\,\mathrm{d}p. \tag{6.3}$$

Furthermore the first inequality is strict except in the trivial case where $\Delta(q,p)$ vanishes for each (q,p).

Proof. By using the preceding proposition with $g(x) = x$, we get

$$m_\Delta = \frac{1}{2}\int_{\mathbb{R}^{2d}} \Delta(q,p)(1 - \mathrm{e}^{-\Delta(q,p)})\,\mathrm{e}^{-H(q,p)}\,\mathrm{d}q\,\mathrm{d}p. \tag{6.4}$$

The inequality $x(1 - \mathrm{e}^{-x}) > 0$, valid for all real $x \neq 0$, yields the lower bound in (6.3).

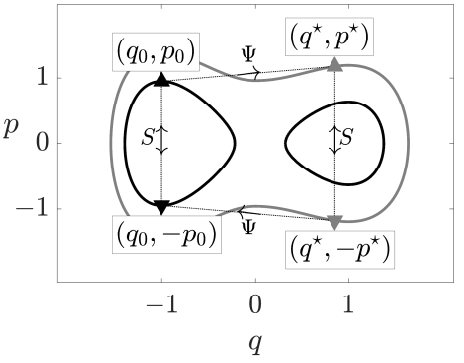


Figure 6.1. A univariate target with probability modes at $q = \pm 1$, leading to a double-well potential $\mathcal{U}$. The continuous lines are contours of constant H. The symmetry of the contours with respect to the axis $p = 0$ is a consequence of the reversibility of the Hamiltonian flow. The solutions of Hamilton's equations move from left to right when $p > 0$ and from right to left when $p < 0$, so when contours are reflected over the horizontal axis the arrow of time is reversed. If a reversible Ψ maps (q_0, p_0) into (q^*, p^*), it has to map $(q^*, -p^*)$ into (q_0, p_0), so as to preserve the symmetry of the figure. The transition $(q_0, p_0) \mapsto (q^*, p^*)$ has an increase in energy and $(q^*, -p^*) \mapsto (q_0, p_0)$ decreases energy in *exactly* the same amount; this is the content of formula (6.1).

For the upper bound, we apply the inequality $|\mathrm{e}^x - 1| \leq |x|(\mathrm{e}^x + 1)$, valid for all $x \in \mathbb{R}$, to obtain

$$\begin{aligned} m_\Delta &\leq \frac{1}{2} \int_{\mathbb{R}^{2d}} |\Delta(q,p)|\, |1 - \mathrm{e}^{-\Delta(q,p)}|\, \mathrm{e}^{-H(q,p)}\, \mathrm{d}q\, \mathrm{d}p \\ &\leq \frac{1}{2} \int_{\mathbb{R}^{2d}} \Delta(q,p)^2 (1 + \mathrm{e}^{-\Delta(q,p)})\, \mathrm{e}^{-H(q,p)}\, \mathrm{d}q\, \mathrm{d}p \\ &\leq \int_{\mathbb{R}^{2d}} \Delta(q,p)^2\, \mathrm{e}^{-H(q,p)}\, \mathrm{d}q\, \mathrm{d}p. \end{aligned}$$

In the last step we used Proposition 6.1 with $g(x) = x^2$. □

Figure 6.1 illustrates the geometry behind formula (6.1). The figure makes clear that to each initial condition (q_0, p_0) with an energy *increase* $\Delta \geq 0$ there corresponds an initial condition $(q^*, -p^*)$ with an energy *decrease* of the same magnitude. In the integrand in (6.2) the energy increase at (q_0, p_0) is weighed by $\exp(-H(q_0, p_0))$, a larger factor than the weight $\exp(-H(q^*, -p^*))$ of the corresponding energy decrease. In addition, by conservation of volume, if (q_0, p_0) ranges in a small domain, then the corresponding points $(q^*, -p^*)$ range in a small domain of the same measure. This explains why the integral (6.2) is positive.

We are of course interested in applying Theorem 6.2 to the case where Ψ is the map Ψ_λ in Step 2 of Algorithm 5.6 (numerical HMC). Assume for simplicity that λ/h is an integer, so that $t = \lambda$ coincides with one of the step points of the numerical integration. Then the energy error satisfies pointwise, that is, at each fixed (q, p),

$$\Delta(p, q) = H(\Psi_\lambda(q, p)) - H(q, p) = [H(\varphi_\lambda(q, p)) + O(h^\nu)] - H(q, p) = O(h^\nu),$$

where ν is the order of the integrator and we have successively used the convergence result in Theorem 3.2 and conservation of energy (Theorem 2.6). The bounds in Theorem 6.2 have important implications.

- *The upper bound.* Even though pointwise the energy error Δ is of size $O(h^\nu)$, *on average* (with respect to the Boltzmann–Gibbs distribution) m_Δ is, at least formally, of size $O(\Delta^2) = O(h^{2\nu})$. The order of the average energy error is automatically *twice* the anticipated order. This is clearly an advantage of using a volume-preserving reversible integrator.
- *The lower bound.* This is a disadvantage. Imagine a case where two uncoupled systems with Hamiltonian functions H_1 and H_2 are juxtaposed. The aggregate is a new Hamiltonian system with Hamiltonian $H_1 + H_2$. The mean energy error for the aggregate is $\mathbb{E}(\Delta_1 + \Delta_2) = \mathbb{E}(\Delta_1) + \mathbb{E}(\Delta_2)$, and because both terms being added are ≥ 0 there is no room for cancellation. In general, the value of the energy and therefore the value of the energy error may be expected to increase as the number of degrees of freedom increases, in agreement with the fact that in physics, energy is an extensive quantity.

This discussion will be continued in Section 7.

6.2. Energy error in the standard Gaussian target

Sections 3.4 and 4.4 were devoted to investigating the behaviour of different integrators when applied to the harmonic oscillator. We take up this theme once more, this time in the HMC context. The aim is then to study what happens when HMC is used to sample from the univariate standard normal, hoping that any findings will be relevant to more complex distributions. (Of course, in practice there is no interest in using an MCMC algorithm to sample from a normal distribution. See the discussion on model problems at the beginning of Section 3.4.)

6.2.1. Pointwise energy error bounds

We return to the situation in Proposition 4.9 and now study the energy error after n time-steps $\Delta(q_0, p_0) = H(p_n, q_n) - H(q_0, p_0)$, where $H = (1/2)(p^2 + q^2)$. The following result (Blanes *et al.* 2014, Proposition 4.2) provides an upper bound for $\Delta(q_0, p_0)$ *uniform* in n.

Proposition 6.3. The energy error, $\Delta(q_0, p_0)$, may be bounded as

$$\Delta(q_0, p_0) \leq \frac{1}{2}(\chi_h^2 - 1)p_0^2$$

if $\chi_h^2 \geq 1$, or as

$$\Delta(q_0, p_0) \leq \frac{1}{2}\left(\frac{1}{\chi_h^2} - 1\right)q_0^2$$

if $\chi_h^2 \leq 1$.

Proof. We only prove the first item; the other is similar. The ellipse (4.10) has its major axis along the co-ordinate axis $p = 0$ of the (q, p)-plane, as in Figure 4.1(a). Hence $2H(q, p) = p^2 + q^2$ attains its maximum on that ellipse if $p = 0$ which implies $q^2 = q_0^2 + \chi_h^2 p_0^2$. If the *final* point (q_n, p_n) of the numerical trajectory happens to be at that maximum, $2\Delta(q_0, p_0) = (q_0^2 + \chi_h^2 p_0^2) - (q_0^2 + p_0^2)$. □

6.2.2. Average energy error bounds

We estimate next the *average* energy error (Blanes *et al.* 2014, Proposition 4.3). Note that the bound provided is once more *uniform* in n.

Proposition 6.4. In the situation described above, assume that (q_0, p_0) is a Gaussian random vector with non-normalized probability density function $\exp(-H(q, p))$, where $H = (1/2)(p^2 + q^2)$. Then the expectation of the random variable $\Delta(q_0, p_0) \in \mathbb{R}^2$ is given by

$$\mathbb{E}(\Delta) = \sin^2(n\theta_h)\,\rho(h),$$

where

$$\rho(h) = \frac{1}{2}\left(\chi_h^2 + \frac{1}{\chi_h^2} - 2\right) = \frac{1}{2}\left(\chi_h - \frac{1}{\chi_h}\right)^2 \geq 0,$$

and accordingly

$$0 \leq \mathbb{E}(\Delta) \leq \rho(h).$$

Proof. With the shorthand $c = \cos(n\theta_h)$, $s = \sin(n\theta_h)$, we may write

$$2\Delta(q_0, p_0) = \left(-\frac{1}{\chi_h}sq_0 + cp_0\right)^2 + (cq_0 + \chi_h sp_0)^2 - (p_0^2 + q_0^2)$$

or

$$2\Delta(q_0, p_0) = s^2\left(\frac{1}{\chi_h^2} - 1\right)q_0^2 + 2cs\left(\chi_h - \frac{1}{\chi_h}\right)q_0p_0 + s^2(\chi_h^2 - 1)p_0^2.$$

Since $\mathbb{E}(q_0^2) = \mathbb{E}(p_0^2) = 1$ and $\mathbb{E}(q_0p_0) = 0$, the proof is complete. □

6.2.3. Energy error bounds for Verlet

Let us illustrate the preceding results in the case of the Verlet integrator. For the velocity version, we find from Example 3.8 and (4.7), for $0 < h < 2$,

$$\chi_h^2 = \frac{h^2}{1-(1-h^2/2)^2} = \frac{1}{1-h^2/4} > 1.$$

The bound in Proposition 6.3 reads

$$\Delta(q_0, p_0) \leq \frac{h^2}{8(1-h^2/4)}\, p_0^2. \tag{6.5}$$

For $h = 1$, $\Delta(q_0, p_0) \leq p_0^2/6$; therefore, if $-2 < p_0 < 2$ (an event that for a standard normal distribution has probability $> 95\%$), then $\Delta(q_0, p_0) < 2/3$, which results in a probability of acceptance $\geq 51\%$, regardless of the number n of time-steps.

The position Verlet integrator (Example 3.9 and Remark 4.10) has $\chi_h^2 = 1 - h^2/4 < 1$ provided that $0 < h < 2$. Proposition 6.3 yields

$$\Delta(q_0, p_0) \leq \frac{h^2}{8(1-h^2/4)}\, q_0^2$$

(as one may have guessed from (6.5) by symmetry).

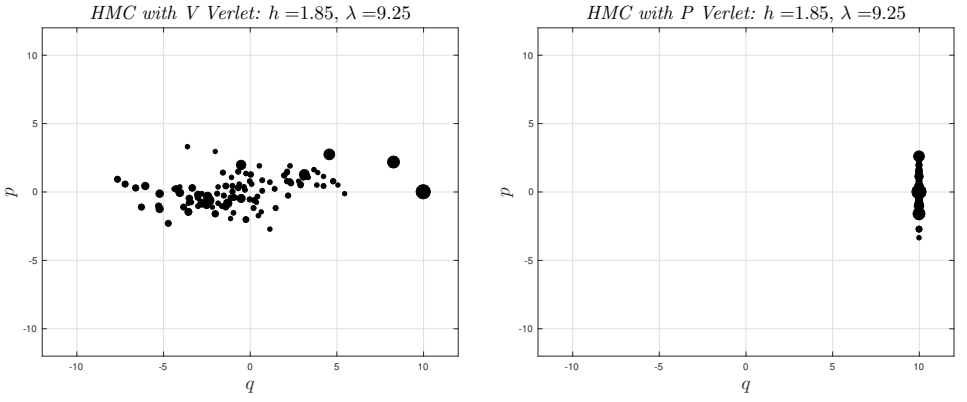
From Proposition 6.4, for both the velocity and the position versions,

$$0 \leq \mathbb{E}(\Delta) \leq \rho(h) = \frac{h^4}{32(1-h^2/4)}. \tag{6.6}$$

We draw attention to the exponent of h in the numerator: *even though, pointwise, for the harmonic oscillator, energy errors for the Verlet integrator are $O(h^2)$, they are $O(h^4)$ on average*, which of course matches our earlier finding in Theorem 6.2. For $h = 1$ the expected energy error is $\leq 1/24$. Halving h to $h = 1/2$ leads to an expected energy error $\leq 1/480$. The conclusion of the examples above is that *acceptance rates for Verlet* are likely to be high even if the step size h is not small.

6.3. Velocity Verlet or position Verlet?

The symmetry between the velocity and position Verlet integrators may lead to the conclusion that their performances in the HMC scenario are equivalent. This is not the case, because in Algorithm 5.6 (for a general target distribution) the q and p variables do not play a symmetric role at all. In particular, samples of p from the correct marginal distribution (Step 1) are used as initial conditions for the integration legs in Step 2. On the contrary, when the Markov chain is initialized, the user has to *guess* a suitable starting value for q (see the end of Section 5.2). In some applications, the chosen starting value may actually correspond to a location of low probability, and we now study the difficulties that may arise in that case.

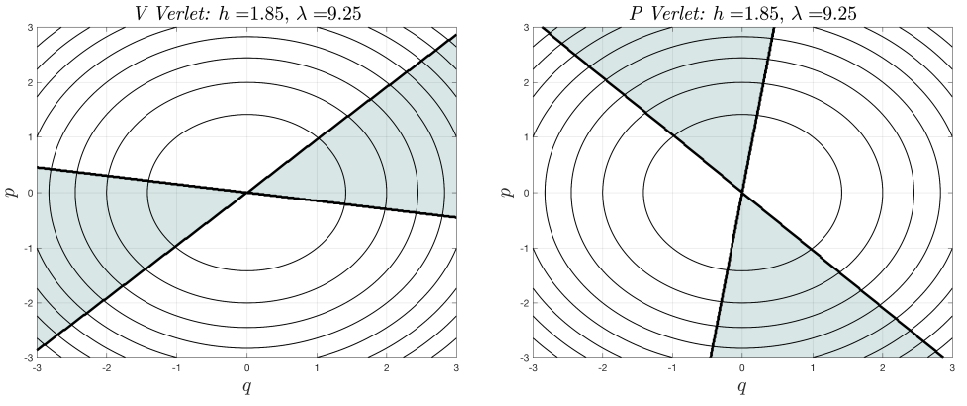


(a) (b)

Figure 6.2. Discrete trajectories of the Markov chain when HMC for the standard Gaussian target is operated with (a) velocity and (b) position Verlet. Each marker is a state of the Markov chain (intermediate values of (q, p) along the numerical trajectories are not depicted). The size of the markers is related to the index i in the chain: states corresponding to larger values of i have smaller markers. The chains start at $q = 10$. In (b), all chain transitions result in rejection, so that $q_i = 10$ for all values of i; position Verlet gets stuck. In (a), the velocity chain quickly identifies the interval around $q = 0$ where the target distribution is concentrated.

Figure 6.2 corresponds to the standard Gaussian/harmonic oscillator target and compares the performance of the velocity and position algorithms when the chain is initialized with q at a location of very low probability, $q_0 = 10$, ten standard deviations away from the mean (the initialization of p is of no consequence since momenta are discarded in the momentum-refreshment Step 1 of Algorithm 5.6). The position integrator gets stuck where its velocity counterpart succeeds without problems.

Insight into the different behaviours of velocity and position Verlet may be gained from Figure 6.3, which corresponds to integrations with the large value $h = 1.85$ carried out in the interval $0 \leq t \leq \lambda = 9.25$ (five time-steps). Regardless of the choice of $|q_0|$, velocity Verlet will result in an energy decrease (and therefore acceptance) if the initial sample ξ_0 of the momentum is such that (q_0, ξ_0) lands in the grey area of panel (a), which will happen with large probability because small values of $|\xi_0|$ are likely to occur. On the contrary, in the position algorithm energy decreases only occur if very large values of ξ_0 are drawn. Certainly, there is symmetry between the grey areas of both panels; however, in HMC, having energy decreases near the q-axis is helpful, and having them near the p-axis is not. It may help to go back to the bound (6.5) which shows that, for velocity Verlet, $p_0 = 0$ ensures an energy decrease; compare both panels in Figure 4.1 as well.



(a) (b)

Figure 6.3. Regions where the energy error $\Delta(q,p)$ is positive/negative for (a) velocity and (b) position Verlet. In the grey-shaded regions $\Delta(q,p)$ is negative, ensuring acceptance of the Markov transition. The contour lines in the background show level sets of $H(q,p)$.

To conclude this discussion we emphasize that if the chains were initialized by drawing samples from the marginal distribution of q (*i.e.* started at stationarity), then the behaviour of velocity and position Verlet would be the same due to the q/p symmetry.

6.4. Multivariate Gaussian model

We study once more the quadratic Hamiltonian (3.27) and note that the covariance matrix Σ of the target is the inverse of the stiffness matrix K. As discussed in Proposition 3.13, in the particular case where $M = I$, the square roots of the eigenvalues of K are the angular frequencies of the dynamics; the eigenvalues of Σ are of course the variances of the target along the directions of the eigenvectors. Thus *small variance implies high frequency.*

6.4.1. Average energy error in the multivariate case

Proposition 6.4 may be extended to the multivariate Gaussian case. We begin by providing a variant of Proposition 3.13.

Proposition 6.5. The change of dependent variables, $q = L^{-T}U\Omega^{-1}Q$, $p = LUP$, decouples the system into a collection of d harmonic oscillators:

$$\frac{\mathrm{d}Q^i}{\mathrm{d}t} = \omega_i P^i, \quad \frac{\mathrm{d}P^i}{\mathrm{d}t} = -\omega_i Q^i, \quad i = 1, \ldots, d$$

(superscripts denote components).

In terms of the new variables, the function in (3.27) is given by

$$\frac{1}{2}P^TP + \frac{1}{2}Q^TQ = \frac{1}{2}\sum_i((P^i)^2 + (Q^i)^2).$$

Therefore the $2d$ variables Q^i and P^i are independent standard Gaussian.[10]

As discussed in connection with Proposition 3.13, we may assume that the numerical method is applied to each of the uncoupled oscillators. After recalling Remark 3.12, the following result is easily proved.

Theorem 6.6. Consider a stable integration of (3.27) with a (consistent), reversible volume-preserving integrator applied with a stable value of h. If (q_0, p_0) are random variables with probability density function (proportional to) $\exp(-H)$, then for each n the expectation of the energy error $\Delta(q_0, p_0) = H(q_n, p_n) - H(q_0, p_0)$ may be bounded as

$$0 \leq \mathbb{E}(\Delta) \leq \sum_{j=1}^{d} \rho(\omega_j h), \tag{6.7}$$

where ρ is the function defined in Proposition 6.4.

6.4.2. An illustration

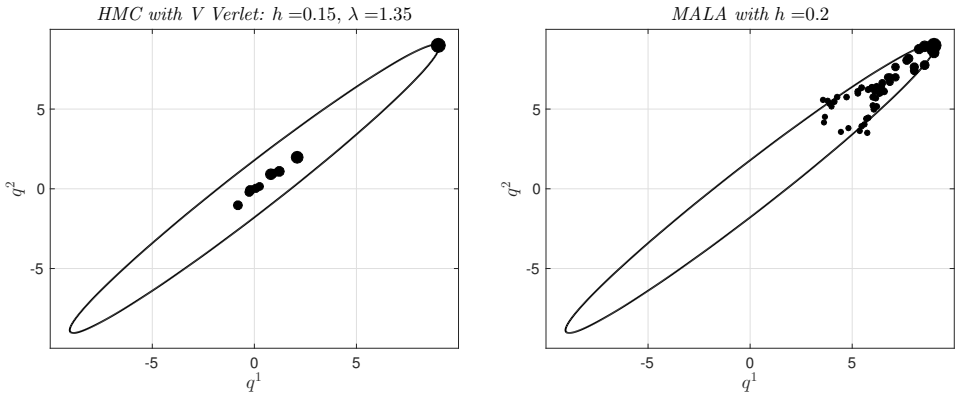
As an example, we consider a two-dimensional target where

$$K = \Sigma^{-1} = \frac{1}{2}\begin{bmatrix} 101 & -99 \\ -99 & 101 \end{bmatrix}.$$

This matrix has eigenvalues $\omega_1^2 = 1$ (with eigenspace $q^1 = q^2$) and $\omega_2^2 = 100$ (with eigenspace $q^1 = -q^2$). The mass matrix is chosen to be $M = I$. We use HMC with velocity Verlet; stability requires $h\omega_2 < 2$ or $h < 0.2$ and we set $h = 0.15$. The chain is started at $(q^1, q^2) = (9, 9)$ where $\mathcal{U}(9, 9) = 81$, that is, about nine standard deviations in the direction of largest variance. The duration parameter λ is set to be equal to 1.35 (*i.e.* there are nine time-steps in each numerical integration leg) and the number of transitions in the chain was the maximum attainable with a computational budget of 100 force evaluations. Figure 6.4(a) shows that HMC rapidly relaxes to a region of high probability.

Remark 6.7. It may be of interest to relate this experiment to the discussion of stiffness in Example 3.14. Here it is the oscillation with frequency ω_1 (*i.e.* the evolution along the diagonal $q^1 = q^2$) that matters most; however,

[10] Obviously, the last display does not give the Hamiltonian function for the Hamiltonian dynamics for (Q, P). Proposition 2.4 does not apply because the change of variables $(q, p) = \Phi(Q, P)$ is not symplectic.



(a) (b)

Figure 6.4. Bivariate Gaussian distribution, (q^1, q^2)-plane; the ellipse is a contour of constant probability density. HMC operated with (a) velocity Verlet and (b) MALA. The size of the markers is related to i: points along the Markov trajectory corresponding to larger values of i have smaller markers. The computational budget for both algorithms is fixed at 100 force evaluations. Along these trajectories, the average acceptance probability for HMC and MALA is 93% and 82%, respectively. The stability requirement for Verlet is $h < 0.2$, which is set by the component of the target distribution with the smallest variance. In relation to MALA, note that HMC relaxes more rapidly to a region of high probability.

h has to be chosen in terms of the oscillation with frequency ω^2.[11] Nevertheless, HMC succeeds because, at each Markov transition, several time-steps are taken.

As a comparison we have also implemented the well-known algorithm MALA, with the same initial state and same computational budget. MALA may be described as the algorithm that results when, in HMC operated with velocity Verlet, the duration parameter λ is chosen to coincide with h, that is, each integration leg only takes one time-step. In this way the accept/reject mechanism operates after each individual time-step in MALA (but only after λ/h successive time-steps in HMC). Figure 6.4(b) shows MALA zigzags in a region of low probability.

Remark 6.8. In connection with the preceding remark, we note that for MALA the progress of the Markov chain along the $q^1 = q^2$ diagonal is slow. Each time-step moves the state by a small amount, because h was chosen by taking into account the behaviour of the problem along the $q^1 = -q^2$ diagonal. The momentum refreshments, which in MALA take place after

[11] Example 3.14 had $\omega_2 = 100$; here we have chosen a smaller value of ω_2 so as not to blur the figure.

every individual time-step, change the direction of motion in the (q^1, q^2)-plane, thus inducing a random walk behaviour. *HMC avoids this random walk behaviour by taking sufficiently many time-steps between consecutive accept/reject steps so that integration legs make substantial strides in the solution components with high variance.*

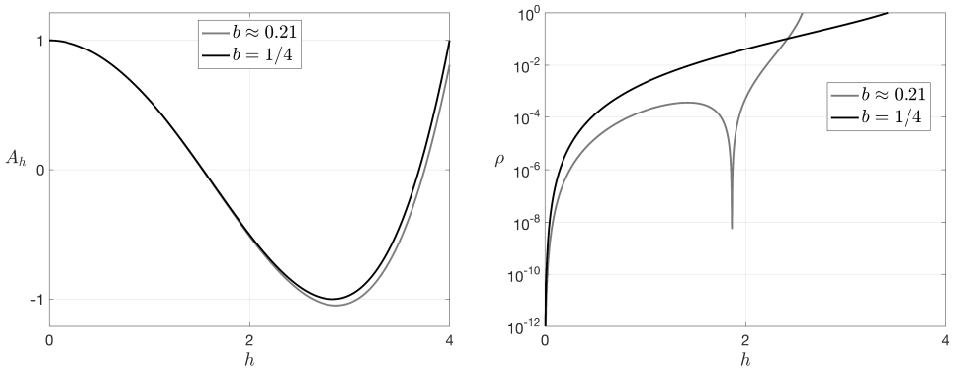
6.5. Can Verlet be beaten?

Currently, velocity Verlet is the integrator of choice within HMC algorithms. Is this the best possibility? In this subsection we will discuss whether it is possible to construct a palindromic splitting formula with $s > 1$ stages (3.19)–(3.20) that improves on the velocity Verlet algorithm in the sense that, when operated with a step size h, leads to smaller energy errors – higher acceptance probabilities – than velocity Verlet with step size h/s. Because, as illustrated above, Verlet is often very successful for large values of the step size, such a construction has to be based on investigating the behaviour of the integrators for finite h and therefore cannot be guided by information (such as the expansion in Theorem 4.6) which corresponds to the behaviour as $h \to 0$. We therefore resort to Gaussian models with Hamiltonian of the form (3.27). In the remainder of this subsection we shall use the symbol τ to refer to the actual value of the step size implemented in the algorithm and keep the letter h for the (non-dimensional) product $\omega\tau$, where ω represents one of the frequencies present in the problem. With this notation, Verlet is stable for $0 < h < 2$ or $0 < \tau < 2/\omega_{\max}$.

When the number d of degrees of freedom equals 1, velocity Verlet is indeed the best choice. We know from Section 6.2 that Verlet will deliver high acceptance rates for h just below its stability limit and, from Section 4.5, that Verlet has the longest stability interval. So Verlet performs well for HMC for values of τ where other integrators are not even stable. Essentially the same is true for d small, as one sees by decoupling the problem into d one-degree-of-freedom oscillators as we did in Section 6.4. However, as d increases, Verlet is not likely to keep high acceptance rates if τ is just below $2/\omega_{\max}$: in fact the additive character of the energy, noted in the discussion of Theorem 6.2, implies that, even if the energy error in each of the one-dimensional oscillators in the problem is small, the energy error for the overall system will likely be large. In that scenario, where, on *accuracy* grounds, Verlet has to be operated with τ significantly smaller than the stability limit $2/\omega_{\max}$, there is room for improvements in efficiency by resorting to more sophisticated integrators, as we discuss next.

6.5.1. Two stages

We begin by studying the one-parameter family (3.17) of methods with two stages (*i.e.* essentially two force evaluations per step). For the choice



(a) (b)

Figure 6.5. Two-stage palindromic integrators (3.17) corresponding to $b = 1/4$ or b given by (6.8). (a) Stability polynomial A_h. For $b = 1/4$ (which provides two steps of velocity Verlet of step length $\tau/2$) stability is lost at $h = 4$ and there is a double root of the equation $A_h = -1$. The method (6.8) has a significantly shorter stability interval because under perturbation the double root of $A_h = -1$ splits into two real roots. (b) The quantity $\rho = \rho(h)$ that governs the expected energy error for Gaussian problems (see (6.7)). When the methods are operated with $0 < h < 2$, *i.e.* $0 < \tau < 2/\omega_{\max}$, the choice (6.8) yields values of ρ much smaller than those corresponding to $b = 1/4$.

$b = 1/4$, one step of length τ of the two-stage method is equivalent to two steps of length $\tau/2$ of velocity Verlet (see equation (4.12) with $N = 2$). The stability polynomial A_h was described in terms of the Chebyshev polynomial T_2 in Section 4.5 and its graph may be seen in Figure 6.5(a). Stability is lost at $h = 4$ when $A_h = 1$; at $h = 2\sqrt{2} \approx 2.82$, there is a double root of the equation $A_h = -1$ (corresponding to the double root at $\zeta = 0$ of the equation $T_2(\zeta) = 2\zeta^2 - 1 = -1$). Small perturbations of $b = 1/4$ turn the double root of the equation $A_h = -1$ into two real roots in the neighbourhood of $h = 2\sqrt{2}$; after such perturbation the length of the stability interval drops from 4 to $\approx 2\sqrt{2}$. Accordingly, in problems where d is small enough for standard velocity Verlet to work well with $\tau\omega_{\max}$ in the interval $(\sqrt{2}, 2)$, no two-stage formula with $b \neq 1/4$ can improve upon Verlet. However, if, when applying Verlet, $\tau\omega_{\max}$ has to be chosen below $\sqrt{2}$ on accuracy grounds, then, as we will see now, efficiency may be improved by choosing b in (3.17) different from 1/4.

Blanes *et al.* (2014) suggest the following procedure to find the value of b. Their numerical experiments show that, for Gaussian problems with $d \leq 1000$, (one-stage) velocity Verlet achieves acceptance rates $\geq 20\,\%$ when $\tau\omega_{\max} \leq 1$ (*i.e.* when τ is less than half of the maximum allowed by stability). They then assume that two-stage integrators should be required

to perform well when $\tau\omega_{\max} \leq 2$. On the other hand, in view of (6.7), for Gaussian models performance is governed by the quantity ρ defined in Proposition 6.4, which for the one-parameter family (3.17) is found to be

$$\rho(h;b) = \frac{h^4(2b^2(1/2-b)h^2 + 4b^2 - 6b + 1)^2}{8(2-bh^2)(2-(1/2-b)h^2)(1-b(1/2-b)h^2)}.$$

Then b is chosen by minimizing

$$\|\rho\|_{(2)} = \max_{0\leq h\leq 2} \rho(h;b),$$

which yields $b = 0.21178\ldots$. To avoid cumbersome decimal expressions, Blanes *et al.* (2014) instead use the approximate value

$$b = \frac{3-\sqrt{3}}{6} \approx 0.21132, \tag{6.8}$$

which gives $\|\rho\|_{(2)} \approx 5 \times 10^{-4}$. For comparison, $b = 1/4$ has a substantially larger $\|\rho\|_{(2)} \approx 4 \times 10^{-2}$. Numerical experiments reported by Blanes *et al.* (2014) confirm that when d is not small the method (6.8) is a clear improvement on Verlet in an example where the target is not Gaussian. See also Mannseth, Kleppe and Skaug (2018).

6.5.2. Three stages

A similar study may be undertaken for the three-stage family (3.18) with two parameters a and b. The choice $a = 1/3$, $b = 1/6$ yields a method consisting of the concatenation of three successive steps of length $\tau/3$ of velocity Verlet (see equation (4.12) with $N = 3$). The graphs of the stability polynomial A_h may be seen in Figure 6.6(a). Stability is lost at $h = 6$ when $A_h = -1$. The double root of $T_3(\zeta) = 4\zeta^3 - 4\zeta = 1$ at $\zeta = -1/2$ gives rise to a double root of $A_h = 1$ at $h = 3\sqrt{3} \approx 5.19$. The double root of $T_3(\zeta) = -1$ gives rise to a double root of $A_h = -1$ at $h = 3$. Perturbing the values $a = 1/3$, $b = 1/6$ leads in general to methods with (short) stability intervals of length ≈ 3, because typically the double root of $A_h = -1$ splits into two single roots. However, it is possible to perturb a and b in such a way that, after perturbation, $A_h = -1$ has a double root close to $h = 3$ as analysed in Campos and Sanz-Serna (2017). For such particular perturbations, stability is lost by crossing the line $A_h = 1$. This is what happens for the choice

$$a = 0.29619504261126, \quad b = 0.11888010966548, \tag{6.9}$$

found in Blanes *et al.* (2014) by minimizing

$$\|\rho\|_{(3)} = \max_{0\leq h\leq 3} \rho(h;b).$$

Note that the range of the maximum is now $0 \leq h \leq 3$ to account for the fact that, when using three force evaluations per step, we aim at integrations

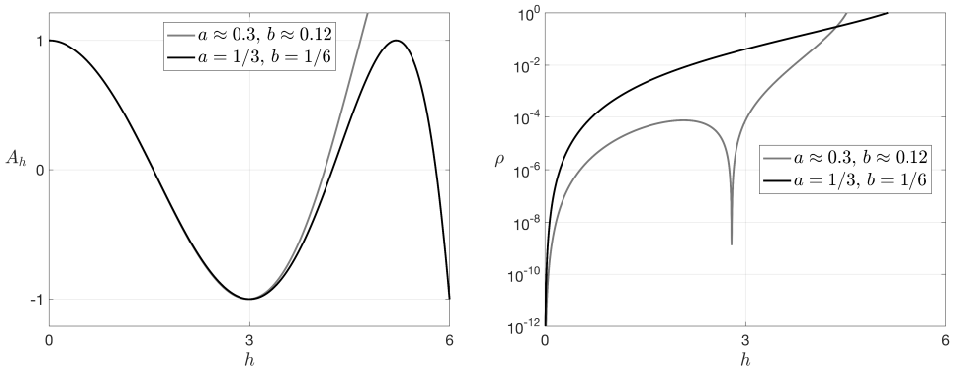


Figure 6.6. Three-stage palindromic integrators (3.18) for $a = 1/3$, $b = 1/6$ or a, b given by (6.9). (a) Stability polynomial A_h. (b) The quantity $\rho = \rho(h)$, which governs the expected energy error for Gaussian problems (see (6.7)). When the methods are operated with $0 < h < 3$, the choice (6.9) yields values of ρ much smaller than those corresponding to $a = 1/3$, $b = 1/6$.

with $\tau\omega_{\max} \leq 3$ to be competitive with Verlet. The choice (6.9) results in $\|\rho\|_{(3)} \approx 7 \times 10^{-5}$ and a stability interval of length $\approx 4.67\ldots$. Blanes *et al.* (2014) report experiments showing the superiority of (6.9) over standard velocity Verlet and also describe the construction of optimized four-stage integrators.

6.5.3. The AIA approach

Let us go back to the one-parameter family (3.17) of methods with two stages. The choice (6.8) is based on the assumption that with $b = 1/4$ the integrator (equivalent to velocity Verlet) would be used with $\tau\omega_{\max}$ in the interval $(0, 2)$. There is of course a degree of arbitrariness in the choice of this interval. When an interval $(0, c)$ with c slightly above 2 is considered, then minimization of

$$\max_{0 \leq h \leq c} \rho(h; b) \tag{6.10}$$

results in a value of b slightly above that corresponding to $(0, 2)$. This improves the length of the stability interval, but, with the notation in Theorem 4.6, increases the quantity $C_{2,1}^2 + C_{2,2}^2$, so that the integrator becomes less accurate in the limit $h \to 0$ (note that $C_{1,1} = 0$ because we are dealing with methods of order 2). As c is increased further, the value of b that minimizes (6.10) increases towards $b = 1/4$; the stability interval improves and accuracy worsens. For $c \geq 2\sqrt{2}$ the optimization procedure leads to $b = 1/4$ because, as pointed out earlier, then $b = 1/4$ is the only value for which the maximum in (6.10) is finite. On the other hand, as c decreases from 2,

the optimal value of b decreases, accuracy in the limit $h \to 0$ improves and the stability interval becomes shorter. In the limit $c \downarrow 0$, b approaches $b \approx 0.1932$, the value that minimizes $C_{2,1}^2 + C_{2,2}^2$ (McLachlan 1995).

In the adaptive integration approach (AIA) suggested by Fernández-Pendás, Akhmatskaya and Sanz-Serna (2016) and Akhmatskaya, Fernández-Pendás, Radivojević and Sanz-Serna (2017) for molecular dynamics problems, the value of b in (3.17) is chosen by minimizing (6.10) with the parameter c adapted to the problem being solved, rather than being kept at the somewhat arbitrary value $c = 2$. The procedure is as follows. In molecular dynamics packages such as GROMACS (Hess, Kutzner, van der Spoel and Lindahl 2008), the user is prompted to specify her choice of the value of τ; of course, smaller values of τ lead to more expensive simulations. The package estimates the values of the frequencies of all harmonic forces present in the problem and will not run if $\tau\tilde{\omega}_{\max}$ is close to the Verlet stability limit ($\tilde{\omega}_{\max}$ is the maximum of the estimated frequencies). AIA sets $c = \sqrt{2}\tilde{\omega}_{\max}\tau$ ($\sqrt{2}$ is a safety factor), so that in (6.10) h ranges in the shortest interval that contains all products $\sqrt{2}\tau\tilde{\omega}_i$, where $\tilde{\omega}_i$ are the estimated frequencies. (If $c \geq 4$, the user is asked to reduce τ.) In this way, whenever the user is prepared to operate with a value of τ that is 'small' for the problem being tackled, AIA chooses a more accurate integrator; as τ increases, AIA will pick up a value of b leading to a more stable, less accurate integrator. This strategy has been successfully implemented in an in-house version of GROMACS. The computational overheads are negligible as they only stem from finding the value of b for the problem at hand and the value of τ chosen by the user. Numerical experiments show that AIA is a clear improvement on Verlet.

7. HMC in high dimension: tuning the step size

We now follow Beskos *et al.* (2013) and study the behaviour of HMC as the dimensionality increases. As a by-product we obtain a general rule for tuning the value of h when running HMC with a chosen value of the duration λ of the integration legs. The analysis uses a model situation similar to that employed in Roberts, Gelman and Gilks (1997) and Roberts and Rosenthal (1998) to analyse other sampling techniques. High-dimensionality in a different setting is the object of the next section.

7.1. The set-up

We consider a high-dimensional Hamiltonian system in $(\mathbb{R}^{2d})^m$, $m \gg 1$, obtained by juxtaposing without coupling m copies of a fixed Hamiltonian system in $\mathbb{R}^{2d}$. If we write a point $(q, p) \in (\mathbb{R}^{2d})^m$ in components as

$$q = (q_1, \ldots, q_m) \in (\mathbb{R}^d)^m \quad \text{and} \quad p = (p_1, \ldots, p_m) \in (\mathbb{R}^d)^m,$$

then the Hamiltonian of the system in $(\mathbb{R}^{2d})^m$ is

$$H_m(q,p) = \sum_{j=1}^{m} H(q_j, p_j),$$

with

$$H(q_j, p_j) = \frac{1}{2} p_j^T M^{-1} p_j + \mathcal{U}(q_j) - \log(Z).$$

Here $\mathcal{U} : \mathbb{R}^d \to \mathbb{R}$ is the potential energy function for each component and Z is the normalizing factor for the one-component Boltzmann–Gibbs distribution, so that $\exp(-H(q_j, p_j))$ is a (normalized) probability density function in $\mathbb{R}^{2d}$.[12] In this way $\exp(-H_m(q,p))$ is the (normalized) probability density function of the Boltzmann–Gibbs distribution in $(\mathbb{R}^{2d})^m$; clearly the $2m$ random vectors q_j, p_j are stochastically independent for this distribution.

From a sampling viewpoint, the target Π is the distribution in $(\mathbb{R}^d)^m$ with non-normalized density $\exp(-\sum_{j=1}^{m} \mathcal{U}(q_j))$; under this target the q_j are *independent and identically distributed.* We sample from Π by using Algorithm 5.6 (HMC), evolving (q,p) by integrating numerically the dynamics associated with H_m. Since there is no coupling between components, in Step 2 of the algorithm each pair (q_j, p_j) moves independently of all the others. In particular, the step size stability restriction for the whole system will coincide with the restriction for the one-component Hamiltonian and will therefore be independent of m. Similarly, in Step 1 randomizing the momentum p in $(\mathbb{R}^d)^m$ is equivalent to randomizing each p_j. However, in Step 3, the different components come together: acceptance depends on the error in H_m and this is obtained by *adding* the errors in the energies of the individual components. As a consequence of Theorem 6.2, for a fixed value of h and a fixed integration interval $0 \le t \le \lambda$, the mean energy error in one component is positive and therefore the mean energy error for H_m will grow linearly with m; as a result, the acceptance rate will decrease as the dimensionality increases. Therefore, for m large with λ fixed, the value of h will have to be decreased to ensure a satisfactory acceptance rate; note that this restriction on h is due to accuracy considerations and has no relation to the stability limit of the integrator being used, which, as we just pointed out, is independent of m in our setting.

7.2. Decreasing h as the dimensionality increases

In what follows we fix the duration λ of the integration leg, assume that λ/h is an integer and study the way in which h has to be decreased if we

[12] The presence of Z in the expression for H simplifies several formulas below but has no consequence in the HMC algorithm itself (the constant $\log(Z)$ does not change the Hamiltonian dynamics and drops from the expression of the acceptance probability).

wish to ensure that the acceptance rate remains bounded away from 0 as $m \uparrow \infty$. We let $\Delta(q_j, p_j; h)$ denote the one component energy error at $t = \lambda$ for an integration started from the initial state (q_j, p_j); this initial state is regarded as a random variable distributed according to the Boltzmann–Gibbs associated with H. Thus (q, p) will be distributed according to the Boltzmann–Gibbs distribution with density $\exp(-H_m)$; in other words the chain is assumed to be at stationarity.

7.2.1. The mean energy error for one component

We shall be concerned with the first and second moments

$$\mu(h) = \int_{\mathbb{R}^{2d}} \Delta(q_j, p_j; h)\, \mathrm{e}^{-H(q_j,p_j)}\, \mathrm{d}q_j\, \mathrm{d}p_j,$$
$$s^2(h) = \int_{\mathbb{R}^{2d}} \Delta(q_j, p_j; h)^2\, \mathrm{e}^{-H(q_j,p_j)}\, \mathrm{d}q_j\, \mathrm{d}p_j,$$

and the corresponding variance $\sigma^2(h) = s^2(h) - \mu(h)^2$.

The following important result rounds out Theorem 6.2. We now assume that $\Delta(q_j, p_j; h)$ behaves asymptotically as $h^\nu a(p_j, q_j)$ for a suitable function a (*cf.* Remark 3.3 where global errors rather than energy errors were considered). In this way, smaller values of $|a|$ correspond to more accurate integrators and/or 'easier' Hamiltonians. The theorem below ensures that the expectation $\mu(h)$ of Δ is $\approx (\Sigma/2)h^{2\nu}$, where we emphasize that the exponent of h is twice the order of the method and that the proportionality constant Σ is the average of a^2. The hypotheses in the theorem will hold in practice under reasonable hypotheses on the integrator and the target distribution; the reader is referred to Beskos *et al.* (2013) for a complete study in the case of the velocity Verlet scheme.

Theorem 7.1. Assume the following.

- There exist functions $a(p_j, q_j)$ and $r(q_j, p_j; h)$ such that
$$\Delta(q_j, p_j; h) = h^\nu a(p_j, q_j) + h^\nu r(q_j, p_j; h),$$
with $\lim_{h\to 0} r(q_j, p_j; h) = 0$ for each (q_j, p_j).
- There exists a real-valued function $D(q_j, p_j)$, integrable with respect to the Boltzmann–Gibbs distribution, such that, for a suitable $h_0 > 0$,
$$\sup_{0<h<h_0} \frac{\Delta(q_j, p_j; h)^2}{h^{2\nu}} \le D(q_j, p_j).$$

Then

$$\lim_{h\to 0} \frac{\mu(h)}{h^{2\nu}} = \frac{\Sigma}{2}, \quad \lim_{h\to 0} \frac{\sigma^2(h)}{h^{2\nu}} = \Sigma,$$

with

$$\Sigma = \int_{\mathbb{R}^{2d}} a(q_j, p_j)^2 \, \mathrm{e}^{-H(q_j,p_j)} \, \mathrm{d}q_j \, \mathrm{d}p_j.$$

Proof. We first establish the second limit. For fixed (q_j, p_j), the first hypothesis implies that $\Delta(q_j, p_j; h)/h^{2\nu} \to a(q_j, p_j)$ and then, by dominated convergence,

$$\lim_{h\to 0} \frac{s^2(h)}{h^{2\nu}} = \int_{\mathbb{R}^{2d}} a(q_j, p_j)^2 \, \mathrm{e}^{-H(q_j,p_j)} \, \mathrm{d}q_j \, \mathrm{d}p_j = \Sigma.$$

In addition, the bound (6.3) shows that

$$\lim_{h\to 0} \frac{\mu(h)^2}{h^{2\nu}} = 0$$

and the limit for $\sigma^2(h)/h^{2\nu}$ follows.

From (6.4), with the shorthand $\Delta = \Delta(q_j, p_j; h)$,

$$\begin{aligned} &\frac{2\mu(h) - \sigma^2(h)}{h^{2\nu}} - \frac{\mu(h)^2}{h^{2\nu}} \\ &\quad = -\int_{\mathbb{R}^{2d}} \frac{\Delta}{h^\nu} \frac{\exp(-\Delta) - 1 + \Delta}{h^\nu} \, \mathrm{e}^{-H(q_j,p_j)} \, \mathrm{d}q_j \, \mathrm{d}p_j. \end{aligned}$$

We note that the second fraction in the integral approaches 0 as $h \to 0$ for fixed (q_j, p_j), because the numerator is $O(\Delta^2)$. A dominated convergence argument (see Beskos *et al.* 2013 for details) shows that the integral also approaches 0 and the proof is complete. □

7.2.2. The mean energy error for m components

For the Hamiltonian H_m the energy error $\Delta_m(q, p; h)$ is given by the sum $\sum_{j=1}^m \Delta(q_j, p_j; h)$. Under our hypotheses, the random variables being added are independent and identically distributed and therefore $\Delta_m(q, p; h)$ has expectation $m\mu(h) \approx m\Sigma h^{2\nu}/2$ and variance $m\sigma^2(h) \approx m\Sigma h^{2\nu}$; thus to ensure that $\Delta_m(q, p; h)$ has a distributional limit as $m \uparrow \infty$, it is reasonable to impose a relation

$$h = \ell m^{-1/(2\nu)}, \tag{7.1}$$

with $\ell > 0$ a constant. Under this relation, a central limit theorem (see Beskos *et al.* 2013 for details) ensures that, as $m \uparrow \infty$, the distribution of the random variable $\Delta_m(q, p; h)$ converges to the distribution of a random variable $\Delta_\infty \sim N(\ell^{2\nu}\Sigma/2, \ell^{2\nu}\Sigma)$. It then follows that the expectation of the acceptance probability $\min\{1, \exp(-\Delta_m)\}$ converges to

$$\mathbb{E}(\min\{1, \mathrm{e}^{-\Delta_\infty}\}).$$

The last expectation may be found analytically and turns out to be

$$A(\ell) = 2\Phi(-\ell^\nu \sqrt{\Sigma}/2), \tag{7.2}$$

where Φ is the standard normal cumulative distribution function

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} \exp\left(-\frac{\xi^2}{2}\right) \mathrm{d}\xi.$$

We then conclude the following.

Theorem 7.2. Assume that the hypotheses of Theorem 7.1 are fulfilled. If in the scenario above the number of copies m and the step size h are related as in (7.1), then, at stationarity, the expectation of the acceptance probability of the HMC algorithm converges to (7.2) as $m \uparrow \infty$.

For Verlet, with $\nu = 2$, the scaling (7.1) entails that halving h compensates for a multiplication of m by a factor of 16. This scaling in high dimension is very favourable when compared with the situation for other sampling techniques (Roberts *et al.* 1997, Roberts and Rosenthal 1998).

7.3. Optimal tuning

In Theorem 7.1, a lower value of Σ, *i.e.* a lower mean value of the function a^2, indicates a more accurate integrator and/or an 'easier' Hamiltonian. Correspondingly, formula (7.2) is such that lowering the value of Σ increases the expected acceptance probability A ($\Phi(x)$ is an increasing function of x). In practice the function a and the constant Σ are unknown, and this would seem to imply that (7.2) is of no practical value. We shall now show that, on the contrary, this relation provides a way to tune the parameter h in the HMC algorithm.

Increasing the value of ℓ in (7.1) leads to a larger step size and therefore lowers the computational cost of each integration leg over the fixed interval $[0, \lambda]$ but typically increases the energy error. Note in this connection that in (7.2), A decreases as ℓ increases. What is the best choice of ℓ? Beskos *et al.* (2013) argue that the function $E(\ell) = \ell A(\ell)$ is a sensible indicator of the efficiency of the algorithm, as its reciprocal measures the amount of work necessary to generate an accepted proposal. With this metric, ℓ should be determined to maximize $\ell A(\ell)$: a direct maximization is not feasible because, as we just noted, we cannot compute $A(\ell)$. This difficulty may be circumvented by treating A as an independent variable and expressing ℓ as a function of A: then

$$E(A) = \frac{2^{1/\nu}}{\Sigma^{1/(2\nu)}} A(\Phi^{-1}(1 - A/2))^{1/\nu}.$$

Clearly the value of A that maximizes E is independent of Σ and is therefore *independent of the target*. In addition the optimal A is the same for all integrators sharing the same value of ν. For the case $\nu = 2$, it is found numerically that

$$A(\Phi^{-1}(1 - A/2))^{1/2}$$

is maximized when

$$A \approx 0.651.$$

This analysis suggests that, for an integrator of order $\nu = 2$ and once the integration interval has been chosen, if m is large, h should be selected in such a way that, in the simulations, the acceptance rate is observed to be close to 65%. If the observed acceptance rate is larger, one would do better with a larger value of h. That would imply more rejections, but the waste caused by the rejections would be offset by the larger number of proposals that may be generated with a fixed budget of force evaluations. Conversely, acceptance rates significantly below 65% indicate that one would do better by working more to generate each single proposal. While our analysis has been performed in the case where the target is a product of m independent identical distributions, practical experience shows that this rule works more generally (Beskos *et al.* 2013).

8. HMC for path sampling: sampling from a perturbed Gaussian distribution

So far the samples generated by the HMC algorithm have been vectors $q \in \mathbb{R}^d$. In some applications the samples needed are *paths*. In this section we discuss the use of HMC in those situations. As will be made clear, the material is directly relevant to the problem of sampling from targets that are perturbations of Gaussian distributions.

8.1. A model problem

In order to keep the presentation as simple as possible, we shall initially limit our attention to a model situation; the general case is discussed at the end of the section.

We consider paths $u(s)$, where u is a real-valued function of the variable $s \in [0, S]$. The paths are constrained by the homogeneous Dirichlet conditions $u(0) = u(S) = 0$ and, formally, their distribution is governed by the 'potential energy' functional

$$\mathcal{U}(u) = \int_0^S \left[\frac{1}{2}(\partial_s u(s))^2 + g(s, u(s))\right] \mathrm{d}s. \tag{8.1}$$

If the function $u(s)$ is smooth, then after integration by parts,

$$\mathcal{U}(u) = \int_0^S \left[-\frac{1}{2}u(s)\partial_{ss}u(s) + g(s, u(s))\right] \mathrm{d}s. \tag{8.2}$$

Distributions of this form arise when studying *diffusion bridge* problems (Reznikoff and Vanden-Eijnden 2005, Beskos *et al.* 2008, Hairer, Stuart and Voss 2009).

Example 8.1. Fix a time[13] horizon $S > 0$ and consider the process $\mathsf{X} : [0, S] \to \mathbb{R}$ that solves the Ornstein–Uhlenbeck equation

$$\mathrm{d}\mathsf{X}(s) = -\mathsf{X}(s)\,\mathrm{d}s + \mathrm{d}\mathsf{B}(s),$$

conditioned on both initial and final conditions $\mathsf{X}(0) = 0$, $\mathsf{X}(S) = 0$. Here B is a standard Brownian motion. The law of this diffusion bridge is a probability measure Π on paths u satisfying the boundary conditions that is associated with the functional

$$\frac{1}{2}\int_0^S [(\partial_s u(s))^2 + u(s)^2]\,\mathrm{d}s. \tag{8.3}$$

Remark 8.2. The case where the process is conditioned on $\mathsf{X}(0) = x^-$, $\mathsf{X}(S) = x^+$ may be reduced to the case with homogeneous boundary conditions, by writing $u(s) = \bar{u}(s) + \ell(s)$, where $\ell(s)$ is a smooth function of s with $\ell(0) = x^-$, $\ell(S) = x^+$. After this transformation, the new paths $\bar{u}(s)$ satisfy homogeneous boundary conditions and their distribution corresponds to a functional of the form (8.1).

A precise mathematical description of the meaning of (8.1) and of the associated probability distribution on paths Π will be given later. For the time being, we note that to study the infinite-dimensional problem on a computer it is necessary to introduce a finite-dimensional discretized version, and we turn to presenting a way of performing the discretization.

We use a uniform grid consisting of $d + 2$ grid points:

$$\{s_j = j\Delta s \mid j = 0, \ldots, d+1\}, \quad \Delta s = S/(d+1).$$

The space of paths is then replaced by the finite-dimensional state space $\mathbb{R}^d$; the jth component u_j of an element $\boldsymbol{u} \in \mathbb{R}^d$ is seen as an approximation to $u(s_j)$, $j = 1, \ldots, d$. The functional (8.1) is discretized as

$$\mathcal{U}_d(\boldsymbol{u}) = \Delta s\left(-\frac{1}{2}\boldsymbol{u}^T\boldsymbol{L}\boldsymbol{u} + G_d(\boldsymbol{u})\right), \quad G_d(\boldsymbol{u}) = \sum_{j=1}^{d} g(u_j) \tag{8.4}$$

(*cf.* (8.2)), where the matrix $\boldsymbol{L}$ corresponds to the standard central difference approximation to ∂_{ss} with homogeneous Dirichlet boundary conditions:

$$\boldsymbol{L} = \frac{1}{\Delta s^2}\begin{bmatrix} -2 & 1 & & \\ 1 & \ddots & \ddots & \\ & \ddots & \ddots & 1 \\ & & 1 & -2 \end{bmatrix}. \tag{8.5}$$

[13] In applications, the variable s typically corresponds to *physical* time, as distinct from the *artificial* time t to be used later in the Hamiltonian dynamics that evolves the paths when obtaining samples.

Note that if the vector $\boldsymbol{u}$ contains the grid values of a smooth path u, then $\mathcal{U}_d(\boldsymbol{u}) \to \mathcal{U}(u)$ as $\Delta s \to 0$.

Our task is then to sample from the target Π_d in $\mathbb{R}^d$ with non-normalized density $\exp(-\mathcal{U}_d(\boldsymbol{u}))$. Special attention has to be paid to the increase in dimension d as the discretization becomes more accurate.

8.2. Preconditioned HMC for path sampling

We now set up an HMC algorithm, called *preconditioned HMC* (PHMC), to sample from our target Π_d.

8.2.1. Algorithm description

As expected, we introduce an auxiliary variable $\boldsymbol{p} \in \mathbb{R}^d$ and a Boltzmann–Gibbs distribution Π_{BGd} in $\mathbb{R}^d \times \mathbb{R}^d$ with non-normalized density $\exp(-\mathcal{H}_d(\boldsymbol{u}, \boldsymbol{p}))$. The Hamiltonian $\mathcal{H}_d$ is given by $\Delta s H_d(\boldsymbol{u}, \boldsymbol{p})$, with

$$H_d(\boldsymbol{u}, \boldsymbol{p}) = \frac{1}{2}\boldsymbol{p}^T \boldsymbol{M}^{-1}\boldsymbol{p} - \frac{1}{2}\boldsymbol{u}^T \boldsymbol{L}\boldsymbol{u} + G_d(\boldsymbol{u})$$

(where $\boldsymbol{M}$ is a mass matrix). Of course, the target Π_d is the $\boldsymbol{u}$-marginal of Π_{BGd}, while the $\boldsymbol{p}$-marginal is Gaussian $N(\boldsymbol{0}, (\Delta s)^{-1}\boldsymbol{M})$.

The pair $(\boldsymbol{u}, \boldsymbol{p})$ is evolved by means of the Hamiltonian dynamics associated with H_d:

$$\begin{bmatrix} \dot{\boldsymbol{u}}(t) \\ \dot{\boldsymbol{p}}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{M}^{-1}\boldsymbol{p}(t) \\ \boldsymbol{L}\boldsymbol{u}(t) - \nabla G_d(\boldsymbol{u}(t)) \end{bmatrix}. \tag{8.6}$$

Clearly, this dynamics preserves the value of $\mathcal{H}_d$ and therefore the distribution Π_{BGd}.

Remark 8.3. Instead of (8.6), one could use the dynamics corresponding to the Hamiltonian $\mathcal{H}_d$ that features in the non-normalized density $\exp(-\mathcal{H}_d(\boldsymbol{u}, \boldsymbol{p}))$ of Π_{BGd}. This alternative dynamics differs from that of (8.6) by a change in scale of t. However, (8.6) is more natural in our context, where there is an infinite-dimensional problem in the background. We illustrate this as follows. Assume that $\boldsymbol{M}$ is taken to be the identity and $g = 0$. Then (8.6), after eliminating $\boldsymbol{p}$, implies $\ddot{\boldsymbol{u}} = \boldsymbol{L}\boldsymbol{u}$, so that $\boldsymbol{u}(t)$ satisfies the standard semidiscrete wave equation. This matches the fact that as $\Delta s \to 0$, $\mathcal{H}_d$ approaches

$$\int_0^S \frac{1}{2}(p(s)^2 + (\partial_s u(s))^2)\, \mathrm{d}s,$$

which provides the Hamiltonian functional (total energy) for the Hamiltonian partial differential equations $\partial_t u = p$, $\partial_t p = \partial_{ss} u$ which, after elimination of p, yield the wave equation $\partial_{tt} u = \partial_{ss} u$.

A challenge with numerically solving (8.6) using an explicit symplectic integrator (like Verlet) is that the spectral radius of $\boldsymbol{L}$ grows with d. Consequently, as the number of grid points increases, the dynamics may become highly oscillatory due to the presence of fast frequencies, and stably approximating this type of dynamics may be difficult (Calvo and Sanz-Serna 2009). For example, numerical stability of a Verlet integrator applied to (8.6) with $G_d = 0$ and $\boldsymbol{M} = \boldsymbol{I}$ requires that its time-step size h be inversely proportional to d.

To avoid this type of restrictive dependence, we *precondition* the dynamics by choosing the mass matrix $\boldsymbol{M} = -\boldsymbol{L}$:

$$\begin{bmatrix} \dot{\boldsymbol{u}}(t) \\ \dot{\boldsymbol{p}}(t) \end{bmatrix} = \begin{bmatrix} -\boldsymbol{L}^{-1}\boldsymbol{p}(t) \\ \boldsymbol{L}\boldsymbol{u}(t) - \nabla G_d(\boldsymbol{u}(t)) \end{bmatrix}. \tag{8.7}$$

In the particular case where g vanishes, we have $\ddot{\boldsymbol{u}} = -\boldsymbol{u}$; all the d frequencies of the preconditioned problem are 1 and Verlet has a step size restriction $h < 2$, *independently* of d.

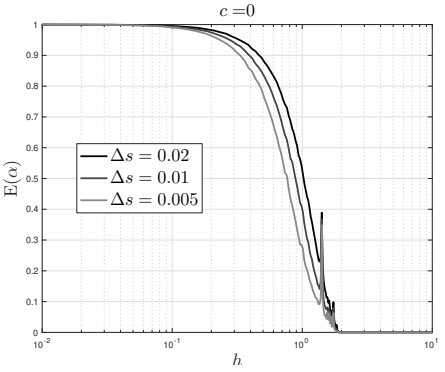
Remark 8.4. Consider for the moment a dynamics of the form (8.6) where $\boldsymbol{L}$ is an arbitrary negative-definite matrix and $\nabla G_d(\boldsymbol{u})$ is small with respect to $\boldsymbol{L}\boldsymbol{u}$. From a sampling viewpoint, this situation would arise if standard HMC were applied to sampling from a perturbation of the centred Gaussian distribution with covariance matrix $-\boldsymbol{L}^{-1}$. The preconditioning $\boldsymbol{M} = -\boldsymbol{L}$ makes sense in that setting. Large eigenvalues of $-\boldsymbol{L}$ correspond to directions in state space with small variances/large forces; the mass in those direction is then chosen to be large so as to ensure small displacements and avoid fast frequencies. This idea may be extended: general targets may be locally approximated by a state-dependent Gaussian model, and one may then choose a state-dependent M as the inverse of the covariance matrix of the Gaussian approximation (Girolami and Calderhead 2011). Unfortunately the state dependence of the mass matrix introduces additional terms in Hamilton's equations, which are no longer of the form (2.9), with the unwelcome consequence that *explicit* volume-preserving reversible integrators do not exist.

In the PHMC algorithm, the system (8.7) is integrated by means of a symplectic reversible scheme with second-order accuracy. We use the Strang splitting (3.13) with

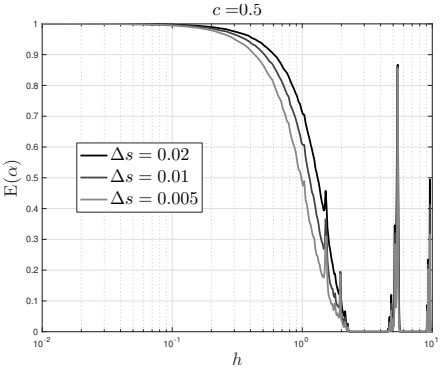
$$\text{(A)} \qquad \begin{bmatrix} \dot{\boldsymbol{u}}(t) \\ \dot{\boldsymbol{p}}(t) \end{bmatrix} = \begin{bmatrix} -\boldsymbol{L}^{-1}\boldsymbol{p}(t) \\ c^2\boldsymbol{L}\boldsymbol{u}(t) \end{bmatrix}$$

and

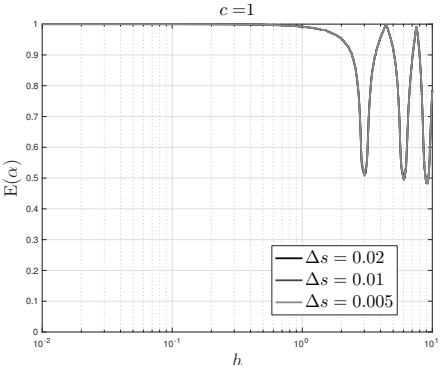
$$\text{(B)} \qquad \begin{bmatrix} \dot{\boldsymbol{u}}(t) \\ \dot{\boldsymbol{p}}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{0} \\ (1-c^2)\boldsymbol{L}\boldsymbol{u}(t) - \nabla G_d(\boldsymbol{u}(t)) \end{bmatrix},$$



(a)



(b)



(c)

Figure 8.1. Ornstein–Uhlenbeck bridge. Mean acceptance probability as a function of h for the PRHMC algorithm for three choices of the splitting parameter c and the three values of Δs. The number of time-steps n at each integration leg is random with a geometric distribution chosen in such a way that the average length $\lambda = nh$ of the integration interval is 20. The number of samples is 10^4. For $c = 1$ the acceptance probability does not decrease as the grid is refined.

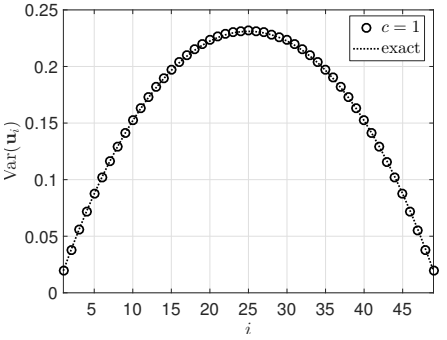


Figure 8.2. Ornstein–Uhlenbeck bridge. This figure assesses the sampling accuracy of the PRHMC algorithm. For the distribution Π_d, the plot graphs the exact and empirical values of the variance of $\boldsymbol{u}_i$ when $\Delta s = 0.02$ corresponding to $d = 49$ interior grid points (the horizontal axis labels the components). The time-step size is $h = 2.0$. The number of time-steps n at each integration leg is random with a geometric distribution chosen in such a way that the average length $\lambda = nh$ of the integration interval is 20. The number of samples is 10^6. The acceptance rate was 95%. When measured in the L^2-norm of $\mathbb{R}^{49}$, the relative error of the vector of empirical variances is 0.36%. The values $c = 0$, $c = 0.5$ were also tested but the acceptance rates were virtually zero.

where $c \in [0, 1]$ is a parameter. The choice $c = 0$ leads to the velocity Verlet method.

Thus, in PHMC, a transition of the HMC Markov chain starts by drawing a new value of the momentum from the marginal distribution

$$N(\mathbf{0}, (\Delta s)^{-1}\boldsymbol{M}) = N(\mathbf{0}, -(\Delta s)^{-1}\boldsymbol{L}),$$

takes $m = \lfloor \lambda/h \rfloor$ steps of the splitting integrator, and accepts/rejects with acceptance probability $\min\{1, \mathrm{e}^{-\Delta \mathcal{H}_d}\}$ (see Algorithm 5.6). A variant called PRMHC, with randomized duration as in Algorithm 5.11, is clearly possible; in that variant the number of steps is chosen to be geometrically distributed with mean λ/h.

Remark 8.5. For implementation purposes, it is advisable to use the variable $\boldsymbol{v} = -\boldsymbol{L}^{-1}\boldsymbol{p}$ rather than $\boldsymbol{p}$. In the variables $(\boldsymbol{u}, \boldsymbol{v})$, the split system (A) takes the trivial form $\dot{\boldsymbol{u}}(t) = \boldsymbol{v}(t)$, $\dot{\boldsymbol{v}}(t) = -c^2\boldsymbol{u}(t)$. Before beginning an integration leg, the fresh value of $\boldsymbol{v}$ is drawn from the corresponding distribution $N(\mathbf{0}, (\Delta s)^{-1}\boldsymbol{L}^{-1})$. When computing the acceptance probability, the Hamiltonian $\mathcal{H}_d$ is correspondingly expressed in terms of $(\boldsymbol{u}, \boldsymbol{v})$.

8.2.2. Numerical illustration

We have implemented the randomized duration PRHMC algorithm in the particular case of the Ornstein–Uhlenbeck bridge in Example 8.1, with $S = 1$

and homogeneous Dirichlet boundary conditions (see Section 9.3 for the case of constant duration PHMC). Note that due to the linearity of the Ornstein–Uhlenbeck process, the target Π_d is Gaussian and there is no need to use MCMC techniques to sample from it; however, using this target provides a convenient test problem. Figure 8.1 plots the mean acceptance probability of PRHMC as a function of h for three values of the parameter c in the integrator and three choices of Δs. Clearly $c = 1$ provides the best performance. Note in particular that for $c = 1$ the acceptance probability is independent of Δs; for the other values of c and fixed h the acceptance probability decreases with Δs. Figure 8.2 shows that the use of $c = 1$ leads to accurate sampling even with h large. The theory in what follows clarifies these numerical results.

8.2.3. Analysis

The results that follow are concerned with the application of PHMC and PRHMC to targets associated with (8.1) in the particular case of the Ornstein–Uhlenbeck bridge (8.3). The proofs are given in Section 9.4.

In the following stability theorem we require the lowest eigenvalue

$$\omega_1^2 = \frac{4}{\Delta s^2} \sin^2\left(\frac{\pi}{2(d+1)}\right) \tag{8.8}$$

of $-\boldsymbol{L}$. It is trivial to show that $2/S \leq \omega_1$; therefore the stability requirements do not become more stringent as $\Delta s \to 0$. This is a consequence of the preconditioning of the dynamics.

Theorem 8.6. If $h > 0$ satisfies

$$\begin{cases} ch + 2\arctan\left(\dfrac{h(1+(1-c^2)\omega_1^2)}{2c\omega_1^2}\right) < \pi & \text{if } c \in (0,1], \\ h < \dfrac{2\omega_1}{\sqrt{1+\omega_1^2}} & \text{if } c = 0, \end{cases} \tag{8.9}$$

then the splitting integrator used in the PHMC and PRMHC algorithms is stable.

We now turn to the mean energy error after an integration leg.

Theorem 8.7. Choose $\eta \in (0, \pi)$ and restrict the attention to step sizes such that $ch \leq \eta$ and the stability requirement (8.9) is satisfied.

- For $c \in [0, 1)$, there exist positive constants $h_0 = h_0(c, \eta)$ and $C = C(c, \eta)$, such that for $h < h_0$, the mean energy error after n time-steps has the bounds

$$0 \leq \mathbb{E}(\Delta \mathcal{H}_d) \leq Cdh^4.$$

- In the case $c = 1$, there exist positive constants $h_0 = h_0(\eta)$ and $C' = C'(\eta)$, such that for $h < h_0$,

$$0 \leq \mathbb{E}(\Delta \mathcal{H}_d) \leq C' h^4.$$

Here, the expected value is over random initial conditions with non-normalized density $\exp(-\mathcal{H}_d(\boldsymbol{u}, \boldsymbol{p}))$.

The exponent of h in the bounds should not be surprising by now. We emphasize that the bounds are *independent* of the number of steps in the integration leg. They therefore apply to the PHMC case where the duration of the integration leg is fixed and to cases where that duration is randomized as in the PRHMC algorithm in the experiments above. For $c \neq 1$ the energy error grows linearly with the number of degrees of freedom. This is the behaviour we found in the scenario studied in Section 7, but we note that here the different components $\boldsymbol{u}_j$ *are not independent nor are they identically distributed* (this point is discussed further in Section 9.4). On the other hand, for $c = 1$ the energy error bound is *uniform* in Δs.

Even though full details will not be given, we mention that the bounds in Theorem 8.7 may be used to derive results on the acceptance rate, in analogy to what we did in Section 7. If $c = 1$, the mean acceptance rate for fixed h is independent of Δs. However, for $c \neq 1$, h has to be scaled as $(\Delta s)^{1/4}$ to ensure that the mean acceptance rate is bounded away from 0 as $\Delta s \to 0$. This agrees with the experiments in Figure 8.1. Note that the scaling $h \sim (\Delta s)^{1/4}$ arises from *accuracy* considerations rather than from *stability* restrictions.

8.3. Hilbert space HMC

The non-normalized density $\propto \exp(-\mathcal{U}_d(\boldsymbol{u}))$ of the target Π_d associated with $\mathcal{U}_d(\boldsymbol{u})$ in (8.4) may be factored as

$$\exp(-\Delta s G_d(\boldsymbol{u})) \times \exp\left(-\frac{\Delta s}{2} \boldsymbol{u}^T \boldsymbol{L} \boldsymbol{u}\right). \tag{8.10}$$

Thus Π_d has non-normalized density

$$\exp(-\Delta s G_d(\boldsymbol{u})) \tag{8.11}$$

with respect to the Gaussian distribution Π_d^0 in $\mathbb{R}^d$ with mean $\mathbf{0}$ and covariance matrix $(\Delta s)^{-1} \boldsymbol{L}^{-1}$. This observation is useful because as $d \uparrow \infty$, Π_d^0 approaches the centred Gaussian distribution Π^0 in the Hilbert space $L^2(0, S)$ with covariance operator $(-\partial_{ss})^{-1}$, where the differential operator $-\partial_{ss}$ has homogeneous boundary conditions (Da Prato and Zabczyk 2014). It is well known that Π^0 is the distribution of the *Brownian bridge in* $[0, S]$

with homogeneous boundary conditions. On the other hand, the first factor in (8.10) is a discretization of

$$\exp\left(-\int_0^S g(s,u(s))\,\mathrm{d}s\right).$$

In this way the path distribution Π may be described as the measure whose density with respect to the Gaussian Π^0 is given by the last display. There is an important difference between the finite- and infinite-dimensional cases. The finite-dimensional Π_d may described in two equivalent ways: (i) as having non-normalized density $\exp(-\mathcal{U}_d(\boldsymbol{u}))$ with respect to the standard Lebesgue measure $\mathrm{d}q$ or (ii) as having non-normalized density (8.11) with respect to the Gaussian Π_d^0. The infinite-dimensional Π cannot be defined by a density with respect to the standard Lebesgue measure in L^2, simply because that measure does not exist. Thus, necessarily Π has to be defined by its density with respect to Π^0.

These considerations lead to studying the general problem of sampling from a target Π defined by a density $\exp(-\Phi(u))$ with respect to a given centred Gaussian reference measure Π_0 in a Hilbert space. The PHMC and PRHMC algorithms described above in the restricted scenario of (8.1) may be extended without difficulty to that general problem. In the extension, the matrix $\boldsymbol{L}$ in (8.5) is replaced by a discretization of the inverse of the covariance operator of Π_0 and the function G_d arises from discretizing Φ. Again the key point in the preconditioned algorithms is to choose the mass matrix $\boldsymbol{M}$ to coincide with $-\boldsymbol{L}$; this ensures that if $\Phi = 0$, the dynamics is given by $\ddot{\boldsymbol{u}} = -\boldsymbol{u}$ where all frequencies are 1.

The general problem just described has been addressed by Beskos *et al.* (2011), who introduced an HMC algorithm that is formulated in *the Hilbert space itself*. Of course, in practice that algorithm can only be implemented after a suitable discretization and, once the discretization has been performed, it *coincides with PHMC with* $c = 1$ implemented as in Remark 8.5. We emphasize that in PHMC we first discretized the target and then formulated the sampling method. In contrast, Beskos *et al.* (2011) proceed in reverse order: discretization comes *after* formulating the sampling method. The route in Beskos *et al.* (2011), which requires non-trivial use of functional-analytic techniques, is mathematically more sophisticated than the approach we have followed here. What is then the advantage of formulating the algorithm in the infinite-dimensional scenario? A sampling method that works in the Hilbert space itself may be expected to work uniformly well as the dimension d of the discretization tends to ∞. This is what happens to the algorithm in Beskos *et al.* (2011) in view of Theorem 8.7. On the other hand, for $c \neq 1$, PHMC cannot arise from discretizing a Hilbert space algorithm, because we know that its performance becomes worse and worse as $d \uparrow \infty$ with h fixed.

To finish this section, we remark that for MALA preconditioned and non-preconditioned versions are available (Beskos *et al.* 2008), and analogous to the latter, there is a non-preconditioned version of HMC (Bou-Rabee 2017).

9. Supplementary material

The paper concludes with material that complements a number of points considered in the preceding sections.

9.1. Finding the modified equation of a splitting integrator

We now show how to obtain the expansion in Theorem 4.6. Full details will be given for the particular case of the Lie–Trotter splitting algorithm (3.11); but the method is the same for more involved integrators. There are two steps. In the first, flows are represented as exponentials. In the second, exponentials are combined by means of the Baker–Campbell–Hausdorff (BCH) formula.

9.1.1. Lie derivatives

Associated with the vector field $f(x)$ in the differential system (2.1) with flow φ_t, there is a *Lie derivative* D_f. This is the first-order differential operator that maps each smooth function $\chi : \mathbb{R}^D \to \mathbb{R}$ into a new function $D_f\chi : \mathbb{R}^D \to \mathbb{R}$ defined as

$$(D_f\chi)(x) = \sum_{i=1}^{D} f^i(x)\frac{\partial}{\partial x^i}\chi(x)$$

(superscripts denote components of vectors). The chain rule leads to the formula

$$\frac{\mathrm{d}}{\mathrm{d}t}\chi(\varphi_t(x))\bigg|_{t=0} = (D_f\chi)(x),$$

which shows the meaning of $(D_f\chi)(x)$ as a rate of change of χ along the solution $t \mapsto \varphi_t(x)$ of (2.1). By successively applying this formula to the functions $D_f\chi, D_f(D_f\chi), \ldots$, we find

$$\frac{\mathrm{d}^k}{\mathrm{d}t^k}\chi(\varphi_t(x))\bigg|_{t=0} = (D_f^k\chi)(x), \quad k = 2, 3, \ldots,$$

where D_f^k is the kth-order differential operator defined inductively as

$$(D_f^k\chi)(x) = (D_f(D_f^{k-1}\chi))(x), \quad k = 2, 3, \ldots.$$

Therefore the Taylor expansion of $\chi(\varphi_t(x))$ at $t = 0$ reads

$$\chi(\varphi_t(x)) = \sum_{k=0}^{\infty} \frac{t^k}{k!}(D_f^k\chi)(x)$$

or

$$\chi(\varphi_t(x)) = (\exp(tD_f)\chi)(x),$$

a formula that may be used to retrieve, at least formally, φ_t: its application with χ equal to the coordinate function $\chi(x) = x^i$, $i = 1, \dots, D$, yields the ith component of the vector $\varphi_t(x)$. In conclusion, the equality

$$\chi \circ \varphi_t = \exp(tD_f)\chi \tag{9.1}$$

may be understood as a representation of the flow of a differential system as the exponential of the Lie operator of its vector field.

If $f^{(\mathrm{A})}$ and $f^{(\mathrm{B})}$ are vector fields, then the compositions $D_{f^{(\mathrm{A})}}D_{f^{(\mathrm{B})}}$ and $D_{f^{(\mathrm{B})}}D_{f^{(\mathrm{A})}}$ are second-order differential operators. However, it is easily checked that $D_{f^{(\mathrm{A})}}D_{f^{(\mathrm{B})}} - D_{f^{(\mathrm{B})}}D_{f^{(\mathrm{A})}}$ is a first-order differential operator. In fact the following result holds (Arnol'd 1989, Section 39).

Proposition 9.1. $D_{f^{(\mathrm{A})}}D_{f^{(\mathrm{B})}} - D_{f^{(\mathrm{B})}}D_{f^{(\mathrm{A})}}$ is the Lie operator corresponding to the vector field $[f^{(\mathrm{A})}, f^{(\mathrm{B})}]$, that is, to the Lie bracket of $f^{(\mathrm{A})}$ and $f^{(\mathrm{B})}$ defined in (2.14).

If now $\varphi_t^{(\mathrm{A})}, \varphi_s^{(\mathrm{B})}$ are the flows corresponding to the vector fields $f^{(\mathrm{A})}$ and $f^{(\mathrm{B})}$ respectively, two applications of (9.1) give

$$\begin{aligned}\chi \circ (\varphi_s^{(\mathrm{B})} \circ \varphi_t^{(\mathrm{A})}) &= (\chi \circ \varphi_s^{(\mathrm{B})}) \circ \varphi_t^{(\mathrm{A})} = \exp(tD_{f^{(\mathrm{A})}})(\chi \circ \varphi_s^{(\mathrm{B})}) \\ &= \exp(tD_{f^{(\mathrm{A})}})(\exp(sD_{f^{(\mathrm{B})}})\chi) \\ &= (\exp(tD_{f^{(\mathrm{A})}})\exp(sD_{f^{(\mathrm{B})}}))\chi.\end{aligned} \tag{9.2}$$

Thus the operator $\exp(tD_{f^{(\mathrm{A})}})\exp(sD_{f^{(\mathrm{B})}})$ represents the composition $\varphi_s^{(\mathrm{B})} \circ \varphi_t^{(\mathrm{A})}$ in analogy with (9.1). Note that the A-flow acts *first* in the composition $\varphi_s^{(\mathrm{B})} \circ \varphi_t^{(\mathrm{A})}$ while $\exp(tD_{f^{(\mathrm{A})}})$ acts *second* in the product of operators $\exp(tD_{f^{(\mathrm{A})}})\exp(sD_{f^{(\mathrm{B})}})$. Our task now is to write $\exp(tD_{f^{(\mathrm{A})}})\exp(sD_{f^{(\mathrm{B})}})$ as a single exponential.

9.1.2. The Baker–Campbell–Hausdorff formula

Assume for the time being that X and Y are square matrices of the same dimension. It is well known that the product $\exp(X)\exp(Y)$ of their exponentials only coincides with $\exp(X + Y)$ if X and Y commute. In fact, by multiplying out

$$\exp(X) = I + X + \frac{1}{2}X^2 + \frac{1}{6}X^3 + \cdots$$

and

$$\exp(Y) = I + Y + \frac{1}{2}Y^2 + \frac{1}{6}Y^3 + \cdots,$$

we find

$$\exp(X)\exp(Y) = I + X + Y + \frac{1}{2}X^2 + XY + \frac{1}{2}Y^2 \\ \frac{1}{6}X^3 + \frac{1}{2}X^2Y + \frac{1}{2}XY^2 + \frac{1}{6}X^3 + \cdots .$$

The products on the right-hand side are all of the form X^kY^ℓ, while the expansion of $\exp(X+Y)$ gives rise to products such as YX, Y^2X, YX^2, XYX, YXY. The BCH formula (Sanz-Serna and Calvo 1994, Hairer *et al.* 2010) writes $\exp(X)\exp(Y)$ as the exponential $\exp(Z)$ of a matrix

$$Z = X + Y + \frac{1}{2}[X,Y] + \frac{1}{12}[X,[X,Y]] + \frac{1}{12}[Y,[Y,X]] \\ + \frac{1}{24}[X,[Y,[Y,X]]] - \frac{1}{720}[Y,[Y,[Y,[Y,X]]]] + \cdots ,$$

where $[\cdot,\cdot]$ denotes the commutator, for example $[X,Y] = XY - YX$. The recipe to write down the terms on the right-hand side is of no consequence for our purposes; what is remarkable is that this right-hand side is a combination of X, Y and *iterated commutators*.

Now the BCH is valid, at least formally (*i.e.* disregarding the convergence of the series involved), if X and Y, instead of matrices, are elements of any associative, non-commutative algebra. In particular it may be applied to the case where X and Y are first-order differential operators, such as those considered above. Going back to (9.2) and recalling that the commutator of the Lie derivatives corresponds to the Lie bracket of the vector fields (Proposition 9.1), the BCH formula with $s = t = h$ then yields

$$\varphi_h^{(\mathrm{B})} \circ \varphi_h^{(\mathrm{A})} = \exp(hD_{\tilde{f}_h^\infty}),$$

where f_h^∞ is the vector field

$$f_h^\infty = f^{(\mathrm{A})} + f^{(\mathrm{B})} + \frac{h}{2}[f^{(\mathrm{A})}, f^{(\mathrm{B})}] + \frac{h^2}{12}[f^{(\mathrm{A})}, [f^{(\mathrm{A})}, f^{(\mathrm{B})}]] + \cdots .$$

Now a comparison with (9.1) shows that $\varphi_h^{(\mathrm{B})} \circ \varphi_h^{(\mathrm{A})}$ is formally the h-flow of f_h^∞, or, in other words, that f_h^∞ is the modified vector-field for the Lie–Trotter splitting algorithm (3.11).

For Strang's method and for more involved splitting integrators the methodology is the same: each of the individual flows whose composition yields ψ_h is written as an exponential and then the exponentials are combined via the BCH formula. For an integrator where ψ_h is the composition of m flows, $m-1$ applications of the BCH are required. The task may be demanding due to the combinatorial intricacies of the BCH formula.

Remark 9.2. As noted in Section 4, finding the modified vector field as above and then resorting to Theorem 4.7 is the most common way of

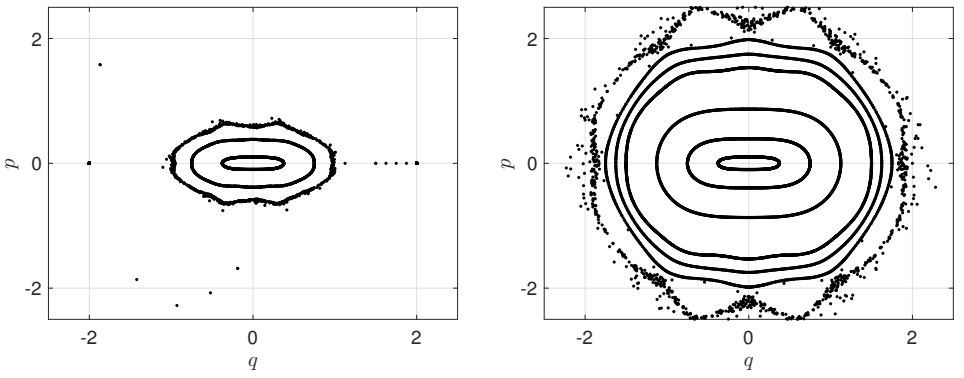


Figure 9.1. Verlet scheme for time-step (a) $h = 1$ and (b) $h = 1/2$ applied to $H(q,p) = (1/2)p^2 + (1/4)q^4$. The numerical solutions mimic the behaviour of the exact solution *only* for initial conditions close to the origin. Decreasing h enlarges the region where the integrator performs satisfactorily, but does not eliminate the problem.

investigating the consistency of splitting algorithms. An alternative direct technique, not based on modified equations, was suggested by Murua and Sanz-Serna (1999). More recently, *word series* (Alamo and Sanz-Serna 2016, Murua and Sanz-Serna 2017) have been introduced as a simpler means to deal with this kind of question. A survey of the combinatorial techniques used to analyse integrators is provided by Sanz-Serna and Murua (2015).

9.2. Distributions with light tails

The numerical integration of Hamiltonian systems with a quadratic Hamiltonian (3.27) (which corresponds to HMC sampling of Gaussian distributions) was discussed in Section 3. For distributions whose tails are lighter than those of a Gaussian, the potential energy $\mathcal{U}(q)$ grows more rapidly than quadratically as $|q| \to \infty$ and this causes difficulties when integrating numerically the Hamiltonian dynamics.

We illustrate this for the univariate distribution with non-normalized density $\exp(-q^3)$. Figure 9.1 shows, for two values of h, nine velocity Verlet trajectories corresponding to initial conditions with $p = 0$ and $q \in (0, 2]$. Whereas for smaller values of the initial q the numerical trajectory mimics the behaviour of the true solution, large initial values of q lead to trajectories that quickly escape to infinity. Reducing the value of h enlarges the size of the domain where Verlet performs satisfactorily. However, no matter how small h is chosen, there will be an outer region where the performance is bad. This is easily understood. The kicks of the Verlet algorithm update p

by using the formula $p \mapsto p - (h/2)q^3$. For this update to be a reasonable approximation to the true dynamics, it is obvious that the magnitude of $(h/2)q^3$ should be small when compared with the magnitude of p; a requirement that does not hold for any fixed h and p if $|q|$ is sufficiently large. The problem is the same for other explicit integrators. For implicit integrators, such as the midpoint rule, the solution of the algebraic equations to be solved at each step also demands smaller values of h for larger values of q as discussed in Sanz-Serna and Calvo (1994, Section 3.3.3).

The conclusion is that, when using an integrator with fixed h, one's attention has to be restricted to a bounded subset D of the phase space $\mathbb{R}^{2d}$ chosen to guarantee that the complement $\mathbb{R}^{2d}\backslash D$ has negligible probability with respect to Π_{BG}. After fixing D, a suitably small value of h has to be chosen. While the situation is well understood for the sampling algorithm MALA (Bou-Rabee and Hairer 2013), the corresponding analysis for HMC has not yet been carried out.

9.3. Randomizing the time-step and the duration

The exact RHMC algorithm, where the length of the integration leg is a random variable with exponential distribution, was discussed at the end of Section 5 along with its numerical counterpart. We also discussed there an alternative procedure, based on randomizing the step length. We now illustrate the performance of these randomized algorithms in the case of the Ornstein–Uhlenbeck bridge example considered in Section 8. With the notation employed there, we set $c = 0$, which leads to the standard Verlet integrator. We have implemented three algorithms: (i) PHMC operated with (constant) step sizes of length h, (ii) PHMC operated with step sizes of length Δt uniformly distributed in the interval $[0.9h, 1.1h]$ and (iii) PRHMC. The results are given in Figure 9.2 (panel (c) is the same as Figure 8.1(a)). As h varies, the behaviour of the non-randomized algorithm (i) is very irregular due to resonances between the step size and the periods of the dynamics (see Section 5.2). Randomizing h regularizes the behaviour, but not as successfully as PRMHC does. This is consistent with the study carried out in Bou-Rabee and Sanz-Serna (2017, §4–5).

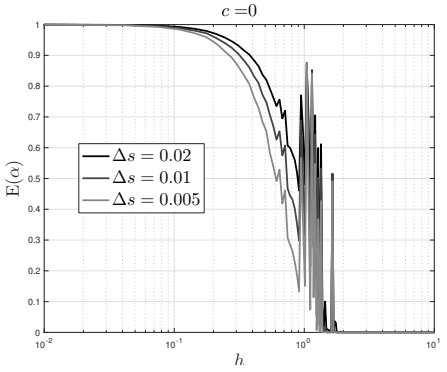
The results of experiments with $c = 0.5$ and $c = 1$, not reported here, show the same patterns.

9.4. PHMC for the Ornstein–Uhlenbeck bridge: proofs

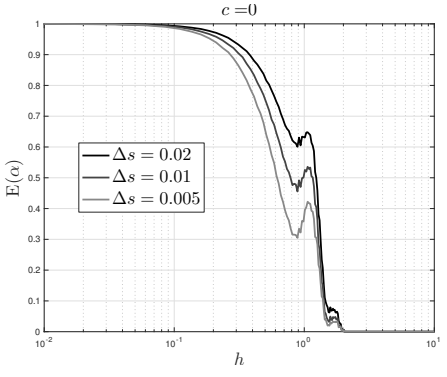
We now prove Theorems 8.6 and 8.7.

9.4.1. One degree of freedom

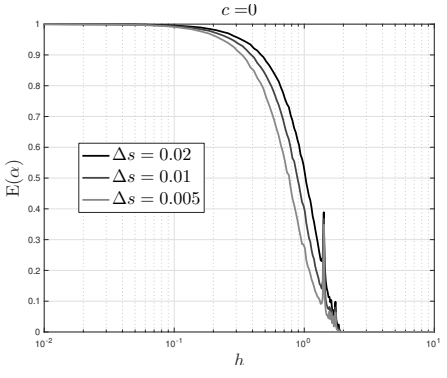
We begin by studying the application of the integrator used in PHMC to a one-degree-of-freedom, linear analogue of the dynamics in (8.7) given by



(a)



(b)



(c)

Figure 9.2. Time-step or duration randomization for the Ornstein–Uhlenbeck bridge. Mean acceptance probability as a function of h for (a) PHMC operated with constant step sizes, (b) PHMC operated with randomized step sizes of average h, and (c) PRHMC, for $c = 0$ and three values of Δs. In all three cases the duration parameter is $\lambda = 20$. The number of samples is 10^4. The PRHMC algorithm is the least susceptible to periodicities in the underlying Hamiltonian dynamics.

the Hamiltonian function

$$H(q,p) = \frac{1}{2}\omega^{-2}p^2 + \frac{1}{2}(\omega^2+1)q^2.$$

If $\omega \gg 1$, this corresponds to the motion of a particle with large mass ω^2 attached to a stiff spring of constant $1+\omega^2$. The solutions are periodic with a frequency $1+1/\omega^2$ that decreases as ω increases.

The one-step propagation matrix (*cf.* (3.22)) is given explicitly by

$$\tilde{M}_{h,c} = \begin{bmatrix} \cos(ch) - \dfrac{h\tilde{\omega}^2}{2c\omega^2}\sin(ch) & \dfrac{\sin(ch)}{c\omega^2} \\ -h\tilde{\omega}^2\cos(ch) - \dfrac{4c^2\omega^4 - h^2\tilde{\omega}^4}{4c\omega^2}\sin(ch) & \cos(ch) - \dfrac{h\tilde{\omega}^2}{2c\omega^2}\sin(ch) \end{bmatrix},$$

with $\tilde{\omega}^2 = 1 + (1-c^2)\omega^2$, if $c \in (0,1]$, or

$$\tilde{M}_{h,c} = \begin{bmatrix} 1 - \dfrac{h^2}{2\omega^2}(1+\omega^2) & \dfrac{h}{\omega^2} \\ \dfrac{h(1+\omega^2)(h^2-(4-h^2)\omega^2)}{4\omega^2} & 1 - \dfrac{h^2}{2\omega^2}(1+\omega^2) \end{bmatrix}$$

if $c = 0$.

The following result is concerned with the stability of the integrator. Note that the stability restriction on h weakens as ω gets larger.

Proposition 9.3. If $\omega > 0$, $c \in [0,1]$, and $h > 0$ satisfies

$$\begin{cases} ch + 2\arctan\left(\dfrac{h(1+(1-c^2)\omega^2)}{2c\omega^2}\right) < \pi & \text{if } c \in (0,1], \\ h < \dfrac{2\omega}{\sqrt{1+\omega^2}} & \text{if } c = 0, \end{cases} \tag{9.3}$$

then the splitting integrator is stable.

Proof. Since the integrator is symplectic, $\det(\tilde{M}_{h,c}) = 1$ and stability is ensured if the magnitude of the trace is < 2. The case $c = 0$ is trivial. For the case $c \in (0,1]$, by means of elementary trigonometric formulas we find

$$\frac{1}{2}\operatorname{trace}(\tilde{M}_{h,c}) = \frac{\cos\left(ch + \arctan\left(\frac{h\tilde{\omega}^2}{2c\omega^2}\right)\right)}{\cos\left(\arctan\left(\frac{h\tilde{\omega}^2}{2c\omega^2}\right)\right)}.$$

As h increases from 0, stability is lost when h is such that the sum of the arguments in the cosine functions reaches π, so that the numerator and the denominator are opposite. □

As in Section 4.4, for stable integrations, $\tilde{M}_{h,c}$ may be written in the form (4.8), with $\cos(\theta_h) = \text{trace}(\tilde{M}_{h,c})/2$ and

$$\chi_h = \begin{cases} \dfrac{\sin(ch)}{c\omega^2 \sin(\theta_h)} & \text{if } c \in (0,1], \\ \dfrac{h}{\omega^2 \sin(\theta_h)} & \text{if } c = 0. \end{cases}$$

For the error Δ in H after an integration leg, by proceeding as in the proof of Proposition 6.4, we may show the following.

Proposition 9.4. For any positive integer n, $\omega > 0$, $c \in [0,1]$, $\beta > 0$ and $h > 0$ satisfying the stability requirement (9.3), the energy error after n steps satisfies

$$\mathbb{E}(\Delta) = \beta^{-1} \sin^2(n\theta_h)\rho(c,\omega,h),$$

where the expectation is over random initial conditions with non-normalized density $\mathrm{e}^{-\beta H(q,p)}$ and

$$\rho(c,\omega,h) = \frac{1}{2}\left(\widehat{\chi}_h^2 + \frac{1}{\widehat{\chi}_h^2} - 2\right) = \frac{1}{2}\left(\widehat{\chi}_h - \frac{1}{\widehat{\chi}_h}\right)^2 \geq 0$$

with

$$\widehat{\chi}_h^2 = (\omega^2 + \omega^4)\chi_h^2.$$

Accordingly

$$0 \leq \mathbb{E}(\Delta) \leq \rho(c,\omega,h).$$

Our next result gives a representation for ρ. The proof is a tedious exercise in algebra and will not be given.

Proposition 9.5. For $\omega > 0$, $c \in [0,1]$, and $h > 0$ satisfying the stability requirement (9.3), we have

$$\rho(c,\omega,h) = \frac{1}{2}\frac{h^4 r^2}{(1+\omega^{-2})(1+\omega^{-2}-h^2 r)}, \tag{9.4}$$

where r is the function

$$r(c,\omega,h) = \frac{1}{4}(1-c^2+\omega^{-2})^2 + c^2(1-c^2+\omega^{-2})R(ch),$$

with

$$R(ch) = \frac{1 - ch\cot(ch)}{(ch)^2}.$$

The function $R(\zeta)$ has the finite limit $1/3$ as $\zeta \to 0$. If $c > 0$, $R(ch)$ is positive and increases monotonically in the interval $0 < ch < \pi$, with

$R(ch) \uparrow \infty$ as $ch \uparrow \pi$. As a consequence, r is an increasing function of $h \in (0, \pi)$; the term $1 + \omega^{-2} - h^2 r$ in (9.4) approaches 0 as h approaches the upper limit of the stability interval. The preceding propositions imply that the mean energy error has an $O(h^4)$ bound. By now this result is hardly a surprise, because we are dealing with an integrator of order $\nu = 2$.

9.4.2. d degrees of freedom

We now address the application of PHMC to the target associated with (8.1) when $g(u) = (1/2)u^2$. Theorems 8.6 and 8.7 are proved by using in $\mathbb{R}^d$ an orthonormal basis of eigenvectors of $-\boldsymbol{L}$ (*cf.* the proof of (6.7)). In the basis of eigenvectors the different components are stochastically independent but not identically distributed.

For Theorem 8.6 it is enough to recall that in Proposition 9.3 larger values of ω lead to less stringent stability requirements. Therefore stability is limited by the smallest eigenvalue of $-\boldsymbol{L}$ given in (8.8).

Turning to the proof of Theorem 8.7, by using the spectral decomposition and Proposition 9.4 with $\beta = \Delta s$, we may write

$$0 \leq \mathbb{E}(\Delta \mathcal{H}_d) \leq \sum_{j=1}^{d} \rho(c, \omega_j, h),$$

with ω_j^2 equal to the jth eigenvalue of $-\boldsymbol{L}$,

$$\omega_j^2 = \frac{4}{\Delta s^2} \sin^2\left(\frac{j\pi}{2(d+1)}\right) \geq \frac{4j^2}{S^2}.$$

We now use (9.4) to obtain the bound

$$0 \leq \mathbb{E}(\Delta \mathcal{H}_d) \leq \frac{h^4}{2} \sum_{j=1}^{d} \frac{r(c, \omega_j, h)^2}{1 - h^2 r(c, \omega_j, h)},$$

and, since r is a decreasing function of ω,

$$0 \leq \mathbb{E}(\Delta \mathcal{H}_d) \leq d \frac{h^4}{2} \frac{r(c, \omega_1, h)^2}{1 - h^2 r(c, \omega_1, h)}.$$

The first bound in the theorem now follows easily: for h bounded away from π, r is bounded, and then h may be reduced further to ensure that $1 - h^2 r(c, \omega_1, h)$ is bounded away from 0.

When $c = 1$,

$$r(1, \omega_j, h) = \frac{1}{4}\omega_j^{-4} + \omega_j^{-2} R(h) \leq \frac{S^4}{64 j^4} + \frac{S^2}{4j^2} R(h),$$

and therefore for h bounded away from π, $r(1, \omega_j, h) = O(j^{-2})$. Thus, if we

write

$$0 \leq \mathbb{E}(\Delta \mathcal{H}_d) \leq \frac{h^4}{2} \sum_{j=1}^{d} \frac{r(1, \omega_j, h)^2}{1 - h^2 r(1, \omega_1, h)},$$

the sum on the right-hand side may be bounded independently of d.

9.5. Convergence

Here is a brief discussion of the sampling properties of HMC.

Intuitively speaking, the point of using HMC is that its Hamiltonian dynamics breaks the random walk behaviour that impairs the convergence of simple MCMC methods like the random walk Metropolis or the MALA algorithm (Gustafson 1998, Diaconis, Holmes and Neal 2000). This slow convergence can be understood by using a diffusion approximation of these simple MCMC methods. In particular, the processes associated with simple MCMC methods are often well approximated (in a weak sense) by overdamped Langevin dynamics (Bou-Rabee *et al.* 2014: see their Theorem 5.1 for a precise mathematical statement), also called Brownian dynamics. This may be seen as a consequence of the detailed balance property of the algorithms. As a result, the corresponding paths meander around state space and explore the target distribution mainly by diffusion. Qualitatively speaking, this is random walk behaviour. The basic idea in HMC is to replace the first-order Langevin dynamics in order to avoid this random walk behaviour.

Unfortunately, beyond this intuition, the improvement in convergence afforded by HMC is not well understood. Under certain conditions on the time-step and target distribution, it has been known for a while that numerical HMC converges to the right distribution asymptotically (Schütte 1999, Stoltz 2007, Cancès, Legoll and Stoltz 2007). More recently, the piecewise deterministic process associated with Algorithm 5.10 (exact RHMC) was shown to be geometrically ergodic, that is, its transition probabilities converge to the target distribution in the total variation metric at an exponential rate (Bou-Rabee and Sanz-Serna 2017, Theorem 3.9). The proof of geometric ergodicity of RHMC relies on Harris's theorem, which requires a (local) version of Doeblin's condition: a minorization condition for the transition probabilities at a finite time and in a compact set. Unfortunately, given the complicated form of these transition probabilities, the minorization condition involves non-explicit constants, and in particular, the dependence of the resulting convergence rate on the mean duration parameter is not clear.

Using a coupling approach, a contraction rate for exact RHMC in a Wasserstein distance has been derived (Bou-Rabee and Eberle 2018). The dependence of the contraction rate on the mean duration parameter is explicit. It follows from this rate that, if the mean duration parameter is proportional to the standard deviation of the target distribution, then the

algorithm converges at a kinetic speed (as opposed to a diffusive speed). The main tools used in the proof are a Markovian coupling and a coupling distance that are both tailored to the structure of exact RHMC. The coupling is based on the framework introduced in Eberle (2016), and is related to a recently developed coupling for second-order Langevin dynamics (Eberle 2018, Eberle, Guillin and Zimmer 2016).

9.6. Related algorithms

We finally discuss generalizations of Algorithm 5.6 (numerical HMC). The corresponding literature is extensive and rapidly growing; we limit ourselves to a few of the many available techniques.

9.6.1. Generalized HMC

Generalized HMC (GHMC) allows the possibility of partial randomization of momentum at every step of the HMC algorithm (Horowitz 1991, Kennedy and Pendleton 2001). Specifically, in GHMC, one introduces a so-called Horowitz angle $\phi \in (0, \pi/2]$ and replaces the randomized momentum ξ_0 in Step 1 of Algorithm 5.6 with $\cos(\phi)p_0 + \sin(\phi)\xi_0$. This partial momentum randomization preserves the p-marginal of Π_{BG}, and hence GHMC leaves Π_{BG} invariant. When $\phi = \pi/2$, the initial momentum p_0 is fully randomized and we recover HMC. If $\phi \in (0, \pi/2)$, then p_0 is only partially randomized; this may be of interest in molecular dynamics applications because then the samples may give more information on the Hamiltonian dynamics than when each integration leg starts with a completely new momentum. The case $\phi = 0$ is excluded, since then there is no momentum randomization, and consequently, the GHMC chain is confined to a single level set of H, and in general, is not ergodic with respect to Π_{BG}. Since the momentum randomizations are the source of dissipation in HMC (Bou-Rabee and Sanz-Serna 2017), partial momentum randomization makes the algorithm less dissipative and less diffusive, as discussed in Neal (2011, §5.5.3). One can similarly introduce a partial momentum randomization to Algorithm 5.11 (numerical RHMC).

9.6.2. Extra chance HMC

As mentioned earlier, rejections increase correlations along the HMC chain and waste computational effort. Extra chance HMC (XHMC) (Campos and Sanz-Serna 2015) provides a simple way for HMC to get extra chances to obtain acceptance, and hence delay or even avert rejection as in Mira (2001) and Green and Mira (2001). To describe the XHMC algorithm, it suffices to describe its accept/reject mechanism, which we state in terms of a symplectic and reversible integrator ψ_h, and its n-step map, $\Psi_\lambda = \psi_h^n$ where $n = \lfloor \lambda/h \rfloor$. Given a user-prescribed number of extra chances K and

$(q,p) \in \mathbb{R}^{2d}$, define $\Gamma^0(q,p) = 0$, $\Gamma^{K+1}(q,p) = 1$, and

$$\Gamma^j(q,p) = \alpha(q,p) \vee \cdots \vee \alpha^j(q,p), \quad 1 \le j \le K,$$

where

$$\alpha^j(q,p) = \min\{1, \mathrm{e}^{-(H(\Psi_\lambda^j(q,p)) - H(q,p))}\}.$$

Note that Γ^j is monotonically increasing with j. Instead of flipping a coin at every step as in Algorithm 5.6, XHMC rolls a $(K+1)$-sided die with probabilities $\{\Gamma^j - \Gamma^{j-1}\}_{j=1}^{K+1}$. Given the current state of the chain $(q_0, p_0) \in \mathbb{R}^{2d}$, the transition in Step 3 of Algorithm 5.6 is accordingly replaced by

$$(q_1, p_1) = \begin{cases} \Psi_\lambda^j(q_0, \xi_0) & \text{if the } j\text{th side of the die comes up,} \\ (q_0, -\xi_0) & \text{if the } (K+1)\text{th side of the die comes up,} \end{cases}$$

where $\xi_0 \sim \mathcal{N}(0, M)$. It is straightforward to show that XHMC leaves Π_{BG} invariant (Campos and Sanz-Serna 2015, Appendix A). Moreover, thanks to the possibility of extra chances, the probability of rejection at every step drops from $\alpha(q_0, p_0)$ to $1 - \Gamma^K(q_0, p_0)$. As its name suggests, the accept/reject mechanism in XHMC can be implemented recursively, so that the proposal move $\Psi_\lambda^j(q_0, p_0)$ and its associated transition probability $\Gamma^j - \Gamma^{j-1}$ are only computed if $\Psi_\lambda^{j-1}(q_0, p_0)$ is not accepted. One can also easily incorporate a partial momentum randomization into XHMC.

9.6.3. Tempering techniques

Probability distributions with multiple modes arise frequently in practice. Different modes may represent metastable conformations of a large molecular system (Frantz, Freeman and Doll 1990, Hansmann and Okamoto 1993, Hansmann 1997, Sugita and Okamoto 1999, Wales 2003) or important features of posterior distributions in Bayesian inference problems (Geyer and Thompson 1995, Neal 1996, Liang and Wong 2001, Ji and Wong 2006, Kou, Zhou and Wong 2006). Unfortunately, HMC may not converge well in situations where the potential energy has multiple minima separated by large barriers. This is a well-known limitation of HMC. It may be due to the low probability of the momentum randomizations in HMC to produce an initial condition whose kinetic energy is sufficiently high that the corresponding solution to Hamilton's equations crosses the potential energy barriers. Also, the momentum randomization may have difficulties producing an initial momentum ξ_0 with the right direction for the solution to Hamilton's equation to exit the current potential well. For such multimodal target distributions, it is advantageous to use HMC in conjunction with a tempering technique such as parallel tempering, simulated tempering, and tempered transition methods (Geyer 1991, Marinari and Parisi 1992, Geyer and Thompson 1995, Neal 1996). Tempering methods

typically introduce an artificial inverse temperature parameter β and apply HMC to the Boltzmann–Gibbs distribution at multiple values of this parameter. The basic idea is that at sufficiently high temperature, the tempered Boltzmann–Gibbs distribution becomes flat enough that barrier crossings in the Hamiltonian integration legs become more likely. Of course, the states of the tempered chains cannot be directly used to compute sample averages with respect to Π_{BG}, which emphasizes that the computational cost of these tempering techniques is by no means negligible.

9.6.4. No U-turns

A practical difficulty with HMC is tuning the durations of the Hamiltonian integration legs. The No-U-Turn Sampler (NUTS) adaptively chooses this parameter by increasing the length of the numerical orbit until the distance between the endpoints of the orbit stops increasing. Unfortunately this procedure breaks reversibility, since, roughly speaking, if this criterion is applied in the reverse direction starting from the end of a trajectory, one does not expect to land at the start of the trajectory. Since reversibility is a key ingredient to the accept/reject mechanism in Algorithm 5.1, and the proof of the associated Theorem 5.7, NUTS uses a different approach: it applies the criterion to the discrete trajectory obtained by integrating the Hamiltonian dynamics in the forward and backward directions. Starting with a single step, the number of forward and backward integration steps is doubled until a U-turn first occurs. NUTS then ends the simulation and samples from all of these points in a way that ensures that the algorithm leaves Π_{BG} invariant. The procedure does not use the Metropolis–Hastings method but rather a variant of the slice sampling method (Neal 2003). Since this sampling step involves all of the states produced by this procedure, the algorithm is thought to have some of the appealing properties of the windowed HMC method, which employs an accept/reject mechanism based on windows of states (Neal 1994, Neal 2011).

9.6.5. Shadow HMC

Given a symplectic and reversible integrator ψ_h for the Hamiltonian dynamics, shadow HMC is an importance sampling technique that replaces the Boltzmann–Gibbs density $\exp(-H)$ with a shadow density $\exp(-\tilde{H}_h^{[\mu]})$ where $\tilde{H}_h^{[\mu]}$ is the modified (or shadow) Hamiltonian of ψ_h up to order μ (Izaguirre and Hampton 2004, Sweet, Hampton, Skeel and Izaguirre 2009, Akhmatskaya and Reich 2008). In so doing, shadow HMC samples from a different target density and the samples have to be reweighed in order to compute averages with respect to the target distribution. This method is not to be confused with the so-called surrogate transition method (Liu 2008), since in shadow HMC the target density is altered. The rationale for shadow

HMC is that $\exp(-\tilde{H}_h)$ is easy to sample from, in the sense that ψ_h preserves $\tilde{H}_h$ much more accurately than H, and hence, for a given duration parameter, shadow HMC increases the acceptance probability without having to reduce the time-step h. There is also a shadow generalized HMC method, which similarly replaces the target density in GHMC with $\exp(-\tilde{H}_h)$. Several aspects of shadow HMC have not been sufficiently analysed from a mathematical viewpoint. For instance the integrability of $\exp(-\tilde{H}_h^{[\mu]})$ has not been studied in depth. Moreover, shadow HMC requires estimating the modified Hamiltonian to order μ, which is non-trivial since it requires computing (or estimating) higher derivatives of $\mathcal{U}$ (Skeel and Hardy 2001). The momentum randomization step in Algorithm 5.6 must be modified, since the p-marginal of $\exp(-\tilde{H}_h^{[\mu]})$ is no longer a Gaussian distribution.

Acknowledgements

This work was supported in part by a Catalyst Grant to N. B-R. from the Provost's Fund for Research at Rutgers University–Camden under project no. 205536, and also in part by the NSF Research Network in Mathematical Sciences: 'Kinetic description of emerging challenges in multiscale problems of natural sciences' (PI, Eitan Tadmor; NSF grant no. 11-07444). J.M.S. has been supported by project MTM2016-77660-P(AEI/FEDER, UE) funded by MINECO (Spain).

REFERENCES[14]

E. Akhmatskaya and S. Reich (2008), 'GSHMC: An efficient method for molecular simulation', *J. Comput. Phys.* **227**, 4937–4954.

E. Akhmatskaya, M. Fernández-Pendás, T. Radivojević and J. M. Sanz-Serna (2017), 'Adaptive splitting integrators for enhancing sampling efficiency of modified Hamiltonian Monte Carlo methods in molecular simulation', in *Tribute to Keith Gubbins, Pioneer in the Theory of Liquids*, special issue of *Langmuir* **33**, 11530–11542.

A. Alamo and J. M. Sanz-Serna (2016), 'A technique for studying strong and weak local errors of splitting stochastic integrators', *SIAM J. Numer. Anal.* **54**, 3239–3257.

M. P. Allen and D. J. Tildesley (1987), *Computer Simulation of Liquids*, Clarendon Press.

C. Andrieu, N. de Freitas, A. Doucet and M. I. Jordan (2003), 'An introduction to MCMC for machine learning', *Machine Learning* **50**, 5–43.

[14] The URLs cited in this work were correct at the time of going to press, but the publisher and the authors make no undertaking that the citations remain live or are accurate or appropriate.

V. I. Arnol'd (1989), *Mathematical Methods of Classical Mechanics* (translated from the Russian by K. Vogtmann and A. Weinstein), Vol. 60 of Graduate Texts in Mathematics, Springer.

S. Asmussen and P. W. Glynn (2007), *Stochastic Simulation: Algorithms and Analysis*, Vol. 57 of Stochastic Modelling and Applied Probability, Springer.

A. Beskos, N. S. Pillai, G. O. Roberts, J. M. Sanz-Serna and A. M. Stuart (2013), 'Optimal tuning of hybrid Monte-Carlo algorithm', *Bernoulli* **19**, 1501–1534.

A. Beskos, F. J. Pinski, J. M. Sanz-Serna and A. M. Stuart (2011), 'Hybrid Monte-Carlo on Hilbert spaces', *Stoch. Proc. Appl.* **121**, 2201–2230.

A. Beskos, G. Roberts, A. Stuart and J. Voss (2008), 'MCMC methods for diffusion bridges', *Stoch. Dynam.* **8**, 319–350.

C. M. Bishop (2006), *Pattern Recognition and Machine Learning*, Springer.

S. Blanes and F. Casas (2016), *A Concise Introduction to Geometric Numerical Integration*, Monographs and Research Notes in Mathematics, CRC Press.

S. Blanes, F. Casas and J. M. Sanz-Serna (2014), 'Numerical integrators for the hybrid Monte Carlo method', *SIAM J. Sci. Comput.* **36**, A1556–A1580.

N. Bou-Rabee (2014), 'Time integrators for molecular dynamics', *Entropy* **16**, 138–162.

N. Bou-Rabee (2017), Cayley splitting for second-order Langevin stochastic partial differential equations. arXiv:1707.05603

N. Bou-Rabee and A. Eberle (2018), Coupling and convergence for exact randomized Hamiltonian Monte-Carlo. In preparation.

N. Bou-Rabee and M. Hairer (2013), 'Non-asymptotic mixing of the MALA algorithm', *IMA J. Numer. Anal.* **33**, 80–110.

N. Bou-Rabee and J. M. Sanz-Serna (2017), 'Randomized Hamiltonian Monte Carlo', *Ann. Appl. Probab.* **27**, 2159–2194.

N. Bou-Rabee and E. Vanden-Eijnden (2010), 'Pathwise accuracy and ergodicity of Metropolized integrators for SDEs', *Comm. Pure Appl. Math.* **63**, 655–696.

N. Bou-Rabee and E. Vanden-Eijnden (2012), 'A patch that imparts unconditional stability to explicit integrators for Langevin-like equations', *J. Comput. Phys.* **231**, 2565–2580.

N. Bou-Rabee, A. Donev and E. Vanden-Eijnden (2014), 'Metropolis integration schemes for self-adjoint diffusions', *Multiscale Model. Simul.* **12**, 781–831.

J. C. Butcher (2016), *Numerical Methods for Ordinary Differential Equations*, third edition, Wiley.

M. P. Calvo and J. M. Sanz-Serna (1993), 'The development of variable-step symplectic integrators, with application to the two-body problem', *SIAM J. Sci. Comput.* **14**, 936–952.

M. P. Calvo and J. M. Sanz-Serna (2009), 'Instabilities and inaccuracies in the integration of highly oscillatory problems', *SIAM J. Sci. Comput.* **31**, 1653–1677.

M. P. Calvo, A. Murua and J. M. Sanz-Serna (1994), Modified equations for ODEs. In *Chaotic Numerics* (P. E. Kloeden and K. J. Palmer, eds), Vol. 172 of Contemporary Mathematics, AMS, pp. 63–74.

C. M. Campos and J. M. Sanz-Serna (2015), 'Extra chance generalized hybrid Monte Carlo', *J. Comput. Phys.* **281**, 365–374.

C. M. Campos and J. M. Sanz-Serna (2017), 'Palindromic 3-stage splitting integrators: A roadmap', *J. Comput. Phys.* **346**, 340–355.

E. Cancès, F. Legoll and G. Stoltz (2007), 'Theoretical and numerical comparison of some sampling methods for molecular dynamics', *Math. Model. Numer. Anal.* **41**, 351–389.

B. Cano and J. M. Sanz-Serna (1997), 'Error growth in the numerical integration of periodic orbits, with application to Hamiltonian and reversible systems', *SIAM J. Numer. Anal.* **34**, 1391–1417.

B. Cano and J. M. Sanz-Serna (1998), 'Error growth in the numerical integration of periodic orbits by multistep methods, with application to reversible systems', *IMA J. Numer. Anal.* **18**, 57–75.

B. Carpenter, A. Gelman, M. Hoffman, D. Lee, B. Goodrich, M. Betancourt, M. A. Brubaker, J. Guo, P. Li and A. Riddell (2016), 'Stan: A probabilistic programming language', *J. Statist. Softw.* **20**, 1–37.

Z. Chen (2003), 'Bayesian filtering: From Kalman filters to particle filters, and beyond', *Statistics* **182**, 1–69.

B. A. Cipra (2000), 'The best of the 20th century: Editors name top 10 algorithms', *SIAM News* **33**(4).

G. Da Prato and J. Zabczyk (2014), *Stochastic Equations in Infinite Dimensions*, Cambridge University Press.

P. Diaconis (2009), 'The Markov chain Monte Carlo revolution', *Bull. Amer. Math. Soc.* **46**, 179–205.

P. Diaconis, S. Holmes and R. M. Neal (2000), 'Analysis of a nonreversible Markov chain sampler', *Ann. Appl. Probab.* **10**, 726–752.

S. Duane, A. D. Kennedy, B. J. Pendleton and D. Roweth (1987), 'Hybrid Monte-Carlo', *Phys. Lett.* B **195**, 216–222.

A. Eberle (2016), 'Reflection couplings and contraction rates for diffusions', *Probab. Theory Rel. Fields* **166**, 851–886.

A. Eberle (2018), A coupling approach to the kinetic Langevin equation on the torus. In preparation.

A. Eberle, A. Guillin and R. Zimmer (2016), Couplings and quantitative contraction rates for Langevin dynamics. arXiv:1703.01617

G. Evensen (2009), *Data Assimilation: The Ensemble Kalman Filter*, Springer Science & Business Media.

Y. Fang, J. M. Sanz-Serna and R. D. Skeel (2014), 'Compressible generalized hybrid Monte Carlo', *J. Chem. Phys.* **140**(17), 174108.

M. Fathi (2014), Theoretical and numerical study of a few stochastic models of statistical physics. PhD thesis, Université Pierre et Marie Curie – Paris VI.

M. Fathi, A.-A. Homman and G. Stoltz (2015), 'Error analysis of the transport properties of Metropolized schemes', *ESAIM Proc. Surv.* **48**, 341–363.

K. Feng and M. Qin (2010), *Symplectic Geometric Algorithms for Hamiltonian Systems* (translated and revised from the Chinese original), Zhejiang Science and Technology, Hangzhou, and Springer.

M. Fernández-Pendás, E. Akhmatskaya and J. M. Sanz-Serna (2016), 'Adaptive multi-stage integrators for optimal energy conservation in molecular simulations', *J. Comput. Phys.* **327**, 434–449.

D. D. Frantz, D. L. Freeman and J. D. Doll (1990), 'Reducing quasi-ergodic behavior in Monte Carlo simulations by J-walking: Applications to atomic clusters', *J. Chem. Phys.* **93**, 2769–2784.

D. Frenkel and B. Smit (2002), *Understanding Molecular Simulation: From algorithms to Applications*, second edition, Academic Press.

A. E. Gelfand and A. F. M. Smith (1990), 'Sampling-based approaches to calculating marginal densities', *J. Amer. Statist. Assoc.* **85**(410), 398–409.

S. Geman and D. Geman (1984), 'Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images', *IEEE Trans. Pattern Anal. Machine Intel.* **PAMI-6**, 721–741.

C. J. Geyer (1991), Markov chain Monte Carlo maximum likelihood. In *Computing Science and Statistics: Proceedings of the 23rd Symposium on the Interface* (E. M. Keramides, ed.), Interface Foundation of North America, pp. 156–163.

C. J. Geyer (1992), 'Practical Markov chain Monte Carlo', *Statist. Sci.* **7**, 473–483.

C. J. Geyer and E. A. Thompson (1995), 'Annealing Markov chain Monte Carlo with applications to ancestral inference', *J. Amer. Statist. Assoc.* **90**(431), 909–920.

Z. Ghahramani (2015), 'Probabilistic machine learning and artificial intelligence', *Nature* **521**(7553), 452.

M. Girolami and B. Calderhead (2011), 'Riemann manifold Langevin and Hamiltonian Monte Carlo methods', *J. Royal Statist. Soc.* B **73**, 123–214.

P. J. Green and A. Mira (2001), 'Delayed rejection in reversible jump Metropolis–Hastings', *Biometrika* **88**, 1035–1053.

D. F. Griffiths and D. J. Higham (2010), *Numerical Methods for Ordinary Differential Equations: Initial Value Problems*, Springer Undergraduate Mathematics Series, Springer.

D. F. Griffiths and J. M. Sanz-Serna (1986), 'On the scope of the method of modified equations', *SIAM J. Sci. Statist. Comput.* **7**, 994–1008.

P. Gustafson (1998), 'A guided walk Metropolis algorithm', *Statist. Comput.* **8**, 357–364.

J. D. Hadfield (2010), 'MCMC methods for multi-response generalized linear mixed models: The MCMCglmm R package', *J. Statist. Softw.* **33**, 1–22.

E. Hairer and G. Wanner (2010), *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*, second revised edition, Vol. 14 of Springer Series in Computational Mathematics, Springer.

E. Hairer, C. Lubich and G. Wanner (2010), *Geometric Numerical Integration*, Springer.

E. Hairer, S. P. Nørsett and G. Wanner (1993), *Solving Ordinary Differential Equations I: Nonstiff Problems*, Vol. 8 of Springer Series in Computational Mathematics, Springer.

M. Hairer, A. M. Stuart and J. Voss (2009), 'Sampling conditioned diffusions', *Trends Stoch. Anal.* **353**, 159–186.

U. H. E. Hansmann (1997), 'Parallel tempering algorithm for conformational studies of biological molecules', *Chem. Phys. Lett.* **281**, 140–150.

U. H. E. Hansmann and Y. Okamoto (1993), 'Prediction of peptide conformation by multicanonical algorithm: New approach to the multiple-minima problem', *J. Comput. Chem.* **14**, 1333–1338.

W. K. Hastings (1970), 'Monte-Carlo methods using Markov chains and their applications', *Biometrika* **57**, 97–109.

B. Hess, C. Kutzner, D. van der Spoel and D. Lindahl (2008), 'GROMACS 4: Algorithms for highly efficient, load-balanced, and scalable molecular simulation', *J. Chem. Theory Comp.* **4**, 435–447.

M. D. Homan and A. Gelman (2014), 'The No-U-Turn Sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo', *J. Mach. Learning Res.* **15**, 1593–1623.

A. M. Horowitz (1991), 'A generalized guided Monte-Carlo algorithm', *Phys. Lett.* B **268**, 247–252.

A. Iserles and G. R. W. Quispel (2017), Why geometric integration? arXiv:1602.07755

J. A. Izaguirre and S. S. Hampton (2004), 'Shadow hybrid Monte Carlo: An efficient propagator in phase space of macromolecules', *J. Comput. Phys.* **200**, 581–604.

S. T. Jensen, X. S. Liu, Q. Zhou and J. S. Liu (2004), 'Computational discovery of gene regulatory binding motifs: A Bayesian perspective', *Statist. Sci.* **19**, 188–204.

H. Ji and W. H. Wong (2006), 'Computational biology: Toward deciphering gene regulatory information in mammalian genomes', *Biometrics* **62**, 645–663.

A. D. Kennedy and B. Pendleton (2001), 'Cost of the generalized hybrid Monte Carlo algorithm for free field theory', *Nucl. Phys.* B **607**, 456–510.

K. Kikuchi, M. Yoshida, T. Maekawa and H. Watanabe (1991), 'Metropolis Monte Carlo method as a numerical technique to solve the Fokker–Planck equation', *Chem. Phys. Lett.* **185**, 335–338.

S. C. Kou, Q. Zhou and W. H. Wong (2006), 'Discussion paper: Equi-energy sampler with applications in statistical inference and statistical mechanics', *Ann. Statist.* **34**, 1581–1619.

W. Krauth (2006), *Statistical Mechanics: Algorithms and Computations*, Oxford University Press.

J. S. W. Lamb and J. A. G. Roberts (1998), 'Time-reversal symmetry in dynamical systems: A survey', *Physica* D *Nonlinear Phenomena* **112**, 1–39.

D. P. Landau and K. Binder (2014), *A Guide to Monte Carlo Simulations in Statistical Physics*, Cambridge University Press.

B. Leimkuhler and S. Reich (2004), *Simulating Hamiltonian Dynamics*, Cambridge Monographs on Applied and Computational Mathematics, Cambridge University Press.

T. Lelièvre, M. Rousset and G. Stoltz (2010), *Free Energy Computations: A Mathematical Perspective*, Imperial College Press.

F. Liang and W. H. Wong (2001), 'Real-parameter evolutionary Monte Carlo with applications to Bayesian mixture models', *J. Amer. Statist. Assoc.* **96**(454), 653–666.

W. A. Link and R. J. Barker (2009), *Bayesian Inference: With Ecological Applications*, Academic Press.

J. S. Liu (2008), *Monte Carlo Strategies in Scientific Computing*, second edition, Springer.

D. J. Lunn, A. Thomas, N. Best and D. Spiegelhalter (2000), 'WinBUGS – a Bayesian modelling framework: Concepts, structure, and extensibility', *Statist. Comput.* **10**, 325–337.

D. Lunn, C. Jackson, N. Best, A. Thomas and D. Spiegelhalter (2012), *The BUGS Book: A Practical Introduction to Bayesian Analysis*, CRC press.

D. Lunn, D. Spiegelhalter, A. Thomas and N. Best (2009), 'The BUGS project: Evolution, critique and future directions', *Statist. Medicine* **28**(25), 3049–3067.

P. B. Mackenzie (1989), 'An improved hybrid Monte Carlo method', *Phys. Lett.* B **226**, 369–371.

J. Mannseth, T. S. Kleppe and H. J. Skaug (2018), 'On the application of improved symplectic integrators in Hamiltonian Monte Carlo', *Commun. Statist. Simul. Comput.* **47**, 500–509.

E. Marinari and G. Parisi (1992), 'Simulated tempering: A new Monte Carlo scheme', *Europhys. Lett.* **19**, 451.

J. Marsden and T. Ratiu (1999), *Introduction to Mechanics and Symmetry*, Springer Texts in Applied Mathematics, Springer.

A. D. Martin, K. M. Quinn and J. H. Park (2011), 'MCMCpack: Markov chain Monte Carlo in R', *J. Statist. Softw.* **42**, 1–21.

R. I. McLachlan (1995), 'On the numerical integration of ordinary differential equations by symmetric composition methods', *SIAM J. Sci. Comput.* **16**, 151–168.

N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller and E. Teller (1953),'Equations of state calculations by fast computing machines', *J. Chem. Phys.* **21**, 1087–1092.

A. Mira (2001), 'On Metropolis–Hastings algorithms with delayed rejection', *Metron* **59**, 231–241.

A. Murua and J. M. Sanz-Serna (1999), 'Order conditions for numerical integrators obtained by composing simpler integrators', *R. Soc. Lond. Philos. Trans.* A *Math. Phys. Eng. Sci.* **357**(1754), 1079–1100.

A. Murua and J. M. Sanz-Serna (2017), 'Word series for dynamical systems and their numerical integrators', *Found. Comput. Math.* **17**, 675–712.

R. M. Neal (1994), 'An improved acceptance procedure for the hybrid Monte Carlo algorithm', *J. Comput. Phys.* **111**, 194–203.

R. M. Neal (1996), 'Sampling from multimodal distributions using tempered transitions', *Statist. Comput.* **6**, 353–366.

R. M. Neal (2003), 'Slice sampling', *Ann. Statist.* **31**, 705–741.

R. M. Neal (2011), MCMC using Hamiltonian dynamics. In *Handbook of Markov Chain Monte Carlo* (S. Brooks *et al.*, eds), Chapman & Hall/CRC, pp. 113–162.

A. Patil, D. Huard and C. J. Fonnesbeck (2010), 'PyMC: Bayesian stochastic modelling in Python', *J. Statist. Softw.* **35**, 1.

M. G. Reznikoff and E. Vanden-Eijnden (2005), 'Invariant measures of stochastic partial differential equations and conditioned diffusions', *Comptes Rendus Math.* **340**, 305–308.

G. O. Roberts and J. S. Rosenthal (1998), 'Optimal scaling of discrete approximations to Langevin diffusions', *J. Roy. Statist. Soc. Ser. B* **60**, 255–268.

G. O. Roberts and R. L. Tweedie (1996*a*), 'Exponential convergence of Langevin distributions and their discrete approximations', *Bernoulli* **2**, 341–363.

G. O. Roberts and R. L. Tweedie (1996*b*), 'Geometric convergence and central limit theorems for multidimensional Hastings and Metropolis algorithms', *Biometrika* **1**, 95–110.

G. O. Roberts, A. Gelman and W. R. Gilks (1997), 'Weak convergence and optimal scaling of random walk Metropolis algorithms', *Ann. Appl. Probab.* **7**, 110–120.

J. M. Sanz-Serna (1991), Two topics in nonlinear stability. In *Advances in Numerical Analysis*, Vol. I (W. Light, ed.), Clarendon Press, pp. 147–174.

J. M. Sanz-Serna (1992), Symplectic integrators for Hamiltonian problems: An overview. In *Acta Numerica*, Vol. 1, Cambridge University Press, pp. 243–286.

J. M. Sanz-Serna (1996), Backward error analysis of symplectic integrators. In *Integration Algorithms and Classical Mechanics* (J. E. Marsden *et al.*, eds), Vol. 10 of Fields Institute Communications, AMS, pp. 193–205.

J. M. Sanz-Serna (1997), Geometric integration. In *The State of the Art in Numerical Analysis* (I. S. Duff and G. A. Watson, eds), Vol. 63 of Institute of Mathematics and its Applications Conference Series, Oxford University Press, pp. 121–143.

J. M. Sanz-Serna (2014), Markov chain Monte Carlo and numerical differential equations. In *Current Challenges in Stability Issues for Numerical Differential Equations* (L. Dieci and N. Guglielmi, eds), Vol. 2082 of Lecture Notes in Mathematics, Springer, pp. 39–88.

J. M. Sanz-Serna (2016), 'Symplectic Runge–Kutta schemes for adjoint equations, automatic differentiation, optimal control, and more', *SIAM Rev.* **58**, 3–33.

J. M. Sanz-Serna and M. P. Calvo (1994), *Numerical Hamiltonian Problems*, Chapman & Hall.

J. M. Sanz-Serna and A. Murua (2015), Formal series and numerical integrators: Some history and some new techniques. In *8th International Congress on Industrial and Applied Mathematics*, Higher Education Press, pp. 311–331.

T. Schlick (2002), *Molecular Modeling and Simulation: An Interdisciplinary Guide*, Vol. 21 of Interdisciplinary Applied Mathematics, Springer.

C. Schütte (1999), Conformational dynamics: Modeling, theory, algorithm, and application to biomolecules. Habilitation, Freie Universität Berlin.

R. D. Skeel and D. J. Hardy (2001), 'Practical construction of modified Hamiltonians', *SIAM J. Sci. Comput.* **23**, 1172–1188.

A. Sokal (1997), Monte Carlo methods in statistical mechanics: Foundations and new algorithms. In *Functional Integration: Basics and Applications* (C. Dewitt-Morette and A. Folacci, eds), Springer, pp. 131–192.

G. Stoltz (2007), Some mathematical methods for molecular and multiscale simulation. PhD thesis, École Nationale des Ponts et Chaussées.

G. Strang (1963), 'Accurate partial difference methods I: Linear Cauchy problems', *Arch. Rational Mech. Anal.* **12**, 392–402.

A. M. Stuart (2010), Inverse problems: A Bayesian perspective. In *Acta Numerica*, Vol. 19, Cambridge University Press, pp. 451–559.

Y. Sugita and Y. Okamoto (1999), 'Replica-exchange molecular dynamics method for protein folding', *Chem. Phys. Lett.* **314**, 141–151.

T. J. Sullivan (2015), *Introduction to Uncertainty Quantification*, Vol. 63 of Texts in Applied Mathematics, Springer.

C. R. Sweet, S. S. Hampton, R. D. Skeel and J. A. Izaguirre (2009), 'A separable shadow Hamiltonian hybrid Monte Carlo method', *J. Chem. Phys.* **131**(17), 174106.

S. Thrun, W. Burgard and D. Fox (2005), *Probabilistic Robotics*, MIT press.

L. Tierney (1994), 'Markov chains for exploring posterior distributions', *Ann. Statist.* **22**, 1701–1728.

M. Tuckerman (2010), *Statistical Mechanics: Theory and Molecular Simulation*, Oxford University Press.

D. Wales (2003), *Energy Landscapes: Applications to Clusters, Biomolecules and Glasses*, Cambridge University Press.

A. R. Webb (2003), *Statistical Pattern Recognition*, Wiley.

H. Yoshida (1990), 'Construction of higher order symplectic integrators', *Phys. Lett.* A **150**, 262–268.