about the center of the galaxy. For small-amplitude librations, the libration frequency is $p$, consistent with our assumption that the oscillation frequency is of order $\epsilon^{1/2}$ when $\Phi_{\rm b}$ is of order $\epsilon$. Large-amplitude librations of this kind may account for the rings of material often seen in barred galaxies (page 538).

We may obtain the shape of the orbit from equation (3.152) by using equation (3.156) to eliminate $\dot\varphi_1 = \frac{1}{2}\dot\psi$:

$$R_1 = -\frac{2R_0\Omega_0\dot\varphi_1}{4\Omega_0^2-\kappa_0^2} = \pm\frac{2^{1/2}R_0\Omega_0}{4\Omega_0^2-\kappa_0^2}\sqrt{E_{\rm p}+p^2\cos(2\varphi_1)}. \tag{3.157}$$

We leave as an exercise the demonstration that when $E_{\rm p} \gg p^2$, equation (3.157) describes the same orbits as are obtained from (3.148a) with $C_1 = 0$ and $\Omega \neq \Omega_{\rm b}$.

The analysis of this subsection complements the analysis of motion near the Lagrange points in §3.3.2. The earlier analysis is valid for small oscillations around a Lagrange point of an arbitrary two-dimensional rotating potential, while the present analysis is valid for excursions of any amplitude in azimuth around the Lagrange points $L_4$ and $L_5$, but only if the potential is nearly axisymmetric.

## 3.4 Numerical orbit integration

In most stellar systems, orbits cannot be computed analytically, so effective algorithms for numerical orbit integration are among the most important tools for stellar dynamics. The orbit-integration problems we have to address vary in complexity from following a single particle in a given, smooth galactic potential, to tens of thousands of interacting stars in a globular cluster, to billions of dark-matter particles in a simulation of cosmological clustering. In each of these cases, the dynamics is that of a Hamiltonian system: with $N$ particles there are $3N$ coordinates that form the components of a vector $\mathbf{q}(t)$, and $3N$ components of the corresponding momentum $\mathbf{p}(t)$. These vectors satisfy Hamilton's equations,

$$\dot{\mathbf{q}} = \frac{\partial H}{\partial \mathbf{p}} \quad ; \quad \dot{\mathbf{p}} = -\frac{\partial H}{\partial \mathbf{q}}, \tag{3.158}$$

which can be written as

$$\frac{{\rm d}\mathbf{w}}{{\rm d}t} = \mathbf{f}(\mathbf{w},t), \tag{3.159}$$

where $\mathbf{w} \equiv (\mathbf{q},\mathbf{p})$ and $\mathbf{f} \equiv (\partial H/\partial\mathbf{p}, -\partial H/\partial\mathbf{q})$. For simplicity we shall assume in this section that the Hamiltonian has the form $H(\mathbf{q},\mathbf{p}) = \frac{1}{2}p^2+\Phi(\mathbf{q})$, although many of our results can be applied to more general Hamiltonians. Given a phase-space position $\mathbf{w}$ at time $t$, and a **timestep** $h$, we require an

algorithm—an **integrator**—that generates a new position $\mathbf{w}'$ that approximates the true position at time $t' = t + h$. Formally, the problem to be solved is the same whether we are following the motion of a single star in a given potential, or the motion of $10^{10}$ particles under their mutual gravitational attraction.

The best integrator to use for a given problem is determined by several factors:

- How smooth is the potential? The exploration of orbits in an analytic model of a galaxy potential places fewer demands on the integrator than following orbits in an open cluster, where the stars are buffeted by close encounters with their neighbors.
- How cheaply can we evaluate the gravitational field? At one extreme, evaluating the field by direct summation in simulations of globular cluster with $\gtrsim 10^5$ particles requires $\mathrm{O}(N^2)$ operations, and thus is quite expensive compared to the $\mathrm{O}(N)$ cost of orbit integrations. At the other extreme, tree codes, spherical-harmonic expansions, or particle-mesh codes require $\mathrm{O}(N \ln N)$ operations and thus are comparable in cost to the integration. So the integrator used in an N-body simulation of a star cluster should make the best possible use of each expensive but accurate force evaluation, while in a cosmological simulation it is better to use a simple integrator and evaluate the field more frequently.
- How much memory is available? The most accurate integrators use the position and velocity of a particle at several previous timesteps to help predict its future position. When simulating a star cluster, the number of particles is small enough ($N \lesssim 10^5$) that plenty of memory should be available to store this information. In a simulation of galaxy dynamics or a cosmological simulation, however, it is important to use as many particles as possible, so memory is an important constraint. Thus for such simulations the optimal integrator predicts the future phase-space position using only the current position and gravitational field.
- How long will the integration run? The answer can range from a few crossing times for the simulation of a galaxy merger to $10^5$ crossing times in the core of a globular cluster. Long integrations require that the integrator does not introduce any systematic drift in the energy or other integrals of motion.

Useful references include Press et al. (1986), Hairer, Lubich, & Wanner (2002), and Aarseth (2003).

### 3.4.1 Symplectic integrators

**(a) Modified Euler integrator** Let us replace the original Hamiltonian $H(\mathbf{q}, \mathbf{p}) = \frac{1}{2}p^2 + \Phi(\mathbf{q})$ by the time-dependent Hamiltonian

$$H_h(\mathbf{q}, \mathbf{p}, t) = \tfrac{1}{2}p^2 + \Phi(\mathbf{q})\delta_h(t), \quad \text{where} \quad \delta_h(t) \equiv h \sum_{j=-\infty}^{\infty} \delta(t - jh) \quad (3.160)$$

is an infinite series of delta functions (Appendix C.1). Averaged over a time interval that is long compared to $h$, $\langle H_h \rangle \simeq H$, so the trajectories determined by $H_h$ should approach those determined by $H$ as $h \to 0$.

Hamilton's equations for $H_h$ read

$$\dot{\mathbf{q}} = \frac{\partial H_h}{\partial \mathbf{p}} = \mathbf{p} \quad ; \quad \dot{\mathbf{p}} = -\frac{\partial H_h}{\partial \mathbf{q}} = -\boldsymbol{\nabla}\Phi(\mathbf{q})\delta_h(t). \tag{3.161}$$

We now integrate these equations from $t = -\epsilon$ to $t = h - \epsilon$, where $0 < \epsilon \ll h$. Let the system have coordinates $(\mathbf{q}, \mathbf{p})$ at time $t = -\epsilon$, and first ask for its coordinates $(\overline{\mathbf{q}}, \overline{\mathbf{p}})$ at $t = +\epsilon$. During this short interval $\mathbf{q}$ changes by a negligible amount, and $\mathbf{p}$ suffers a kick governed by the second of equations (3.161). Integrating this equation from $t = -\epsilon$ to $\epsilon$ is trivial since $\mathbf{q}$ is fixed, and we find

$$\overline{\mathbf{q}} = \mathbf{q} \quad ; \quad \overline{\mathbf{p}} = \mathbf{p} - h\boldsymbol{\nabla}\Phi(\mathbf{q}); \tag{3.162a}$$

this is called a **kick step** because the momentum changes but the position does not. Next, between $t = +\epsilon$ and $t = h - \epsilon$, the value of the delta function is zero, so the system has constant momentum, and Hamilton's equations yield for the coordinates at $t = h - \epsilon$

$$\mathbf{q}' = \overline{\mathbf{q}} + h\overline{\mathbf{p}} \quad ; \quad \mathbf{p}' = \overline{\mathbf{p}}; \tag{3.162b}$$

this is called a **drift step** because the position changes but the momentum does not. Combining these results, we find that over a timestep $h$ starting at $t = -\epsilon$ the Hamiltonian $H_h$ generates a map $(\mathbf{q}, \mathbf{p}) \to (\mathbf{q}', \mathbf{p}')$ given by

$$\mathbf{p}' = \mathbf{p} - h\boldsymbol{\nabla}\Phi(\mathbf{q}) \quad ; \quad \mathbf{q}' = \mathbf{q} + h\mathbf{p}'. \tag{3.163a}$$

Similarly, starting at $t = +\epsilon$ yields the map

$$\mathbf{q}' = \mathbf{q} + h\mathbf{p} \quad ; \quad \mathbf{p}' = \mathbf{p} - h\boldsymbol{\nabla}\Phi(\mathbf{q}'). \tag{3.163b}$$

These maps define the "kick-drift" or "drift-kick" **modified Euler integrator**. The performance of this integrator in a simple galactic potential is shown in Figure 3.21.

The map induced by any Hamiltonian is a canonical or symplectic map (page 803), so it can be derived from a generating function. It is simple to confirm using equations (D.93) that the generating function $S(\mathbf{q}, \mathbf{p}') = \mathbf{q}\cdot\mathbf{p}' + \frac{1}{2}hp'^2 + h\Phi(\mathbf{q})$ yields the kick-drift modified Euler integrator (3.163a).

According to the modified Euler integrator, the position after timestep $h$ is

$$\mathbf{q}' = \mathbf{q} + h\mathbf{p}' = \mathbf{q} + h\mathbf{p} - h^2\boldsymbol{\nabla}\Phi(\mathbf{q}), \tag{3.164}$$

while the exact result may be written as a Taylor series,

$$\mathbf{q}' = \mathbf{q} + h\dot{\mathbf{q}}(t=0) + \tfrac{1}{2}h^2\ddot{\mathbf{q}}(t=0) + \mathrm{O}(h^3) = \mathbf{q} + h\mathbf{p} - \tfrac{1}{2}h^2\boldsymbol{\nabla}\Phi(\mathbf{q}) + \mathrm{O}(h^3). \tag{3.165}$$

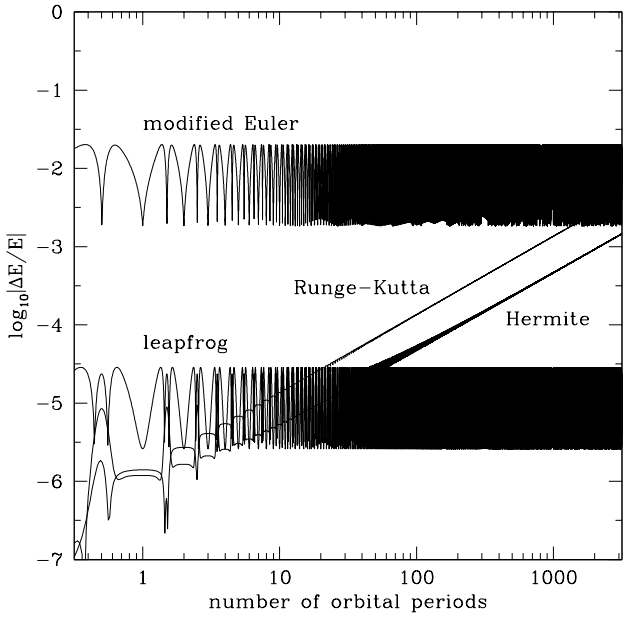


**Figure 3.21** Fractional energy error as a function of time for several integrators, following a particle orbiting in the logarithmic potential $\Phi(r) = \ln r$. The orbit is moderately eccentric (apocenter twice as big as pericenter). The timesteps are fixed, and chosen so that there are 300 evaluations of the force or its derivatives per period for all of the integrators. The integrators shown are kick-drift modified Euler (3.163a), leapfrog (3.166a), Runge–Kutta (3.168), and Hermite (3.172a–d). Note that (i) over moderate time intervals, the errors are smallest for the fourth-order integrators (Runge–Kutta and Hermite), intermediate for the second-order integrator (leapfrog), and largest for the first-order integrator (modified Euler); (ii) the energy error of the symplectic integrators does not grow with time.

The error after a single step of the modified Euler integrator is seen to be $\mathrm{O}(h^2)$, so it is said to be a **first-order** integrator.

Since the mappings (3.163) are derived from the Hamiltonian (3.160), they are symplectic, so either flavor of the modified Euler integrator is a **symplectic integrator**. Symplectic integrators conserve phase-space volume and Poincaré invariants (Appendix D.4.2). Consequently, if the integrator is used to advance a series of particles that initially lie on a closed curve in the $(q_i, p_i)$ phase plane, the curve onto which it moves the particles has the same line integral $\oint p_i \mathrm{d}q_i$ around it as the original curve. This conservation property turns out to constrain the allowed motions in phase space so strongly that the usual tendency of numerical orbit integrations to drift in energy (sometimes called **numerical dissipation**, even though the energy can either decay or grow) is absent in symplectic integrators (Hairer,

Lubich, & Wanner 2002).

**Leapfrog integrator** By alternating kick and drift steps in more elaborate sequences, we can construct higher-order integrators (Yoshida 1993); these are automatically symplectic since they are the composition of maps (the kick and drift steps) that are symplectic. The simplest and most widely used of these is the **leapfrog** or **Verlet** integrator in which we drift for $\frac{1}{2}h$, kick for $h$ and then drift for $\frac{1}{2}h$:

$$\mathbf{q}_{1/2} = \mathbf{q} + \tfrac{1}{2}h\mathbf{p} \ ; \ \mathbf{p}' = \mathbf{p} - h\boldsymbol{\nabla}\Phi(\mathbf{q}_{1/2}) \ ; \ \mathbf{q}' = \mathbf{q}_{1/2} + \tfrac{1}{2}h\mathbf{p}'. \qquad (3.166a)$$

This algorithm is sometimes called "drift-kick-drift" leapfrog; an equally good form is "kick-drift-kick" leapfrog:

$$\mathbf{p}_{1/2} = \mathbf{p} - \tfrac{1}{2}h\boldsymbol{\nabla}\Phi(\mathbf{q}) \ ; \ \mathbf{q}' = \mathbf{q} + h\mathbf{p}_{1/2} \ ; \ \mathbf{p}' = \mathbf{p}_{1/2} - \tfrac{1}{2}h\boldsymbol{\nabla}\Phi(\mathbf{q}'). \quad (3.166b)$$

Drift-kick-drift leapfrog can also be derived by considering motion in the Hamiltonian (3.160) from $t = -\frac{1}{2}h$ to $t = \frac{1}{2}h$.

The leapfrog integrator has many appealing features: (i) In contrast to the modified Euler integrator, it is second- rather than first-order accurate, in that the error in phase-space position after a single timestep is $\mathrm{O}(h^3)$ (Problem 3.26). (ii) Leapfrog is **time reversible** in the sense that if leapfrog advances the system from $(\mathbf{q}, \mathbf{p})$ to $(\mathbf{q}', \mathbf{p}')$ in a given time, it will also advance it from $(\mathbf{q}', -\mathbf{p}')$ to $(\mathbf{q}, -\mathbf{p})$ in the same time. Time-reversibility is a constraint on the phase-space flow that, like symplecticity, suppresses numerical dissipation, since dissipation is not a time-reversible phenomenon (Roberts & Quispel 1992; Hairer, Lubich, & Wanner 2002). (iii) A sequence of $n$ leapfrog steps can be regarded as a drift step for $\frac{1}{2}h$, then $n$ kick-drift steps of the modified Euler integrator, then a drift step for $-\frac{1}{2}h$; thus if $n \gg 1$ the leapfrog integrator requires negligibly more work than the same number of steps of the modified Euler integrator. (iv) Leapfrog also needs no storage of previous timesteps, so is economical of memory.

Because of all these advantages, most codes for simulating collisionless stellar systems use the leapfrog integrator. Time-reversible, symplectic integrators of fourth and higher orders, derived by combining multiple kick and drift steps, are described in Problem 3.27 and Yoshida (1993).

One serious limitation of symplectic integrators is that they work well only with fixed timesteps, for the following reason. Consider an integrator with fixed timestep $h$ that maps phase-space coordinates $\mathbf{w}$ to $\mathbf{w}' = \mathbf{W}(\mathbf{w}, h)$. The integrator is symplectic if the function $\mathbf{W}$ satisfies the symplectic condition (D.78), which involves the Jacobian matrix $g_{\alpha\beta} = \partial W_\alpha / \partial w_\beta$. Now suppose that the timestep is varied, by choosing it to be some function $h(\mathbf{w})$ of location in phase space, so $\mathbf{w}' = \mathbf{W}[\mathbf{w}, h(\mathbf{w})] \equiv \widetilde{\mathbf{W}}(\mathbf{w})$. The Jacobian matrix of $\widetilde{\mathbf{W}}$ is not equal to the Jacobian matrix of $\mathbf{W}$, and in general will not satisfy the symplectic condition; in words, a symplectic integrator with fixed timestep is generally no longer symplectic once the timestep is varied.

Fortunately, the geometric constraints on phase-space flow imposed by time-reversibility are also strong, so the leapfrog integrator retains its good behavior if the timestep is adjusted in a time-reversible manner, even though the resulting integrator is no longer symplectic. Here is one way to do this: suppose that the appropriate timestep $h$ is given by some function $\tau(\mathbf{w})$ of the phase-space coordinates. Then we modify equations (3.166a) to

$$\begin{aligned} \mathbf{q}_{1/2} = \mathbf{q} + \tfrac{1}{2}h\mathbf{p} \quad ; \quad \mathbf{p}_{1/2} = \mathbf{p} - \tfrac{1}{2}h\boldsymbol{\nabla}\Phi(\mathbf{q}_{1/2}), \\ t' = t + \tfrac{1}{2}(h + h'), \\ \mathbf{p}' = \mathbf{p}_{1/2} - \tfrac{1}{2}h'\boldsymbol{\nabla}\Phi(\mathbf{q}_{1/2}) \quad ; \quad \mathbf{q}' = \mathbf{q}_{1/2} + \tfrac{1}{2}h'\mathbf{p}'. \end{aligned} \tag{3.167}$$

Here $h'$ is determined from $h$ by solving the equation $u(h, h') = \tau(\mathbf{q}_{1/2}, \mathbf{p}_{1/2})$, where $\tau(\mathbf{q}, \mathbf{p})$ is the desired timestep at $(\mathbf{q}, \mathbf{p})$ and $u(h, h')$ is any symmetric function of $h$ and $h'$ such that $u(h, h) = h$; for example, $u(h, h') = \frac{1}{2}(h + h')$ or $u(h, h') = 2hh'/(h + h')$.

### 3.4.2 Runge–Kutta and Bulirsch–Stoer integrators

To follow the motion of particles in a given smooth gravitational potential $\Phi(\mathbf{q})$ for up to a few hundred crossing times, the fourth-order Runge–Kutta integrator provides reliable transportation. The algorithm is

$$\begin{aligned} \mathbf{k}_1 = h\mathbf{f}(\mathbf{w}, t) \quad ; \quad \mathbf{k}_2 = h\mathbf{f}(\mathbf{w} + \tfrac{1}{2}\mathbf{k}_1, t + \tfrac{1}{2}h), \\ \mathbf{k}_3 = h\mathbf{f}(\mathbf{w} + \tfrac{1}{2}\mathbf{k}_2, t + \tfrac{1}{2}h) \quad ; \quad \mathbf{k}_4 = h\mathbf{f}(\mathbf{w} + \mathbf{k}_3, t + h), \\ \mathbf{w}' = \mathbf{w} + \tfrac{1}{6}(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4) \quad ; \quad t' = t + h. \end{aligned} \tag{3.168}$$

The Runge–Kutta integrator is neither symplectic nor reversible, and it requires considerably more memory than the leapfrog integrator because memory has to be allocated to $\mathbf{k}_1, \ldots, \mathbf{k}_4$. However, it is easy to use and provides fourth-order accuracy.

The **Bulirsch–Stoer** integrator is used for the same purposes as the Runge–Kutta integrator; although more complicated to code, it often surpasses the Runge–Kutta integrator in performance. The idea behind this integrator is to estimate $\mathbf{w}(t + h)$ from $\mathbf{w}(t)$ using first one step of length $h$, then two steps of length $h/2$, then four steps of length $h/4$, etc., up to $2^K$ steps of length $h/2^K$ for some predetermined number $K$. Then one extrapolates this sequence of results to the coordinates that would be obtained in the limit $K \to \infty$. Like the Runge–Kutta integrator, this integrator achieves speed and accuracy at the cost of the memory required to hold intermediate results. Like all high-order integrators, the Runge–Kutta and Bulirsch–Stoer integrators work best when following motion in smooth gravitational fields.

### 3.4.3 Multistep predictor-corrector integrators

We now discuss more complex integrators that are widely used in simulations of star clusters. We have a trajectory that has arrived at some phase-space position $\mathbf{w}_0$ at time $t_0$, and we wish to predict its position $\mathbf{w}_1$ at $t_1$. The general idea is to assume that the trajectory $\mathbf{w}(t)$ is a polynomial function of time $\mathbf{w}^{\rm poly}(t)$, called the **interpolating polynomial**. The interpolating polynomial is determined by fitting to some combination of the present position $\mathbf{w}_0$, the past positions, $\mathbf{w}_{-1}, \mathbf{w}_{-2}, \ldots$ at times $t_{-1}, t_{-2}, \ldots$, and the present and past phase-space velocities, which are known through $\dot{\mathbf{w}}_j = \mathbf{f}(\mathbf{w}_j, t_j)$. There is no requirement that $\mathbf{f}$ is derived from Hamilton's equations, so these methods can be applied to any first-order differential equations; on the other hand they are not symplectic.

If the interpolating polynomial has order $k$, then the error after a small time interval $h$ is given by the first term in the Taylor series for $\mathbf{w}(t)$ not represented in the polynomial, which is $\mathrm{O}(h^{k+1})$. Thus the order of the integrator is $k$.[12]

The **Adams–Bashforth** multistep integrator takes $\mathbf{w}^{\rm poly}$ to be the unique $k$th-order polynomial that passes through $\mathbf{w}_0$ at $t_0$ and through the $k$ points $(t_{-k+1}, \dot{\mathbf{w}}_{-k+1}), \ldots, (t_0, \dot{\mathbf{w}}_0)$.

Explicit formulae for the Adams–Bashforth integrators are easy to find by computer algebra; however, the formulae are too cumbersome to write here except in the special case of equal timesteps, $t_{j+1} - t_j = h$ for all $j$. Then the first few Adams–Bashforth integrators are

$$\mathbf{w}_1 = \mathbf{w}_0 + h \begin{cases} \dot{\mathbf{w}}_0 & (k=1) \\ \frac{3}{2}\dot{\mathbf{w}}_0 - \frac{1}{2}\dot{\mathbf{w}}_{-1} & (k=2) \\ \frac{23}{12}\dot{\mathbf{w}}_0 - \frac{4}{3}\dot{\mathbf{w}}_{-1} + \frac{5}{12}\dot{\mathbf{w}}_{-2} & (k=3) \\ \frac{55}{24}\dot{\mathbf{w}}_0 - \frac{59}{24}\dot{\mathbf{w}}_{-1} + \frac{37}{24}\dot{\mathbf{w}}_{-2} - \frac{3}{8}\dot{\mathbf{w}}_{-3} & (k=4). \end{cases} \tag{3.169}$$

The case $k = 1$ is called **Euler's integrator**, and usually works rather badly.

The **Adams–Moulton** integrator differs from Adams–Bashforth only in that it computes the interpolating polynomial from the position $\mathbf{w}_0$ and the phase-space velocities $\dot{\mathbf{w}}_{-k+2}, \ldots, \dot{\mathbf{w}}_1$. For equal timesteps, the first few Adams–Moulton integrators are

$$\mathbf{w}_1 = \mathbf{w}_0 + h \begin{cases} \dot{\mathbf{w}}_1 & (k=1) \\ \frac{1}{2}\dot{\mathbf{w}}_1 + \frac{1}{2}\dot{\mathbf{w}}_0 & (k=2) \\ \frac{5}{12}\dot{\mathbf{w}}_1 + \frac{2}{3}\dot{\mathbf{w}}_0 - \frac{1}{12}\dot{\mathbf{w}}_{-1} & (k=3) \\ \frac{3}{8}\dot{\mathbf{w}}_1 + \frac{19}{24}\dot{\mathbf{w}}_0 - \frac{5}{24}\dot{\mathbf{w}}_{-1} + \frac{1}{24}\dot{\mathbf{w}}_{-2} & (k=4). \end{cases} \tag{3.170}$$

[12] Unfortunately, the term "order" is used both for the highest power retained in the Taylor series for $\mathbf{w}(t)$, $t^k$, and the dependence of the one-step error on the timestep, $h^{k+1}$; fortunately, both orders are the same.

Since $\dot{\mathbf{w}}_1$ is determined by the unknown phase-space position $\mathbf{w}_1$ through $\dot{\mathbf{w}}_1 = \mathbf{f}(\mathbf{w}_1, t_1)$, equations (3.170) are nonlinear equations for $\mathbf{w}_1$ that must be solved iteratively. The Adams–Moulton integrator is therefore said to be **implicit**, in contrast to Adams–Bashforth, which is **explicit**.

The strength of the Adams–Moulton integrator is that it determines $\mathbf{w}_1$ by *interpolating* the phase-space velocities, rather than by extrapolating them, as with Adams–Bashforth. This feature makes it a more reliable and stable integrator; the cost is that a nonlinear equation must be solved at every timestep.

In practice the Adams–Bashforth and Adams–Moulton integrators are used together as a **predictor-corrector** integrator. Adams–Bashforth is used to generate a preliminary value $\mathbf{w}_1$ (the prediction or P step), which is then used to generate $\dot{\mathbf{w}}_1 = \mathbf{f}(\mathbf{w}_1, t_1)$ (the evaluation or E step), which is used in the Adams–Moulton integrator (the corrector or C step). This three-step sequence is abbreviated as PEC. In principle one can then iterate the Adams–Moulton integrator to convergence through the sequence PECEC$\cdots$; however, this is not cost-effective, since the Adams–Moulton formula, even if solved exactly, is only an approximate representation of the differential equation we are trying to solve. Thus one usually stops with PEC (stop the iteration after evaluating $\mathbf{w}_1$ twice) or PECE (stop the iteration after evaluating $\dot{\mathbf{w}}_1$ twice).

When these methods are used in orbit integrations, the equations of motion usually have the form $\dot{\mathbf{x}} = \mathbf{v}$, $\dot{\mathbf{v}} = -\boldsymbol{\nabla}\Phi(\mathbf{x}, t)$. In this case it is best to apply the integrator only to the second equation, and to generate the new position $\mathbf{x}_1$ by analytically integrating the interpolating polynomial for $\mathbf{v}(t)$—this gives a formula for $\mathbf{x}_1$ that is more accurate by one power of $h$.

Analytic estimates (Makino 1991) suggest that the one-step error in the Adams–Bashforth–Moulton predictor-corrector integrator is smaller than the error in the Adams–Bashforth integrator by a factor of 5 for $k = 2$, 9 for $k = 3$, 13 for $k = 4$, etc. These analytic results, or the difference between the predicted and corrected values of $\mathbf{w}_1$, can be used to determine the longest timestep that is compatible with a prescribed target accuracy—see §3.4.5.

Because multistep integrators require information from the present time and $k - 1$ past times, a separate startup integrator, such as Runge–Kutta, must be used to generate the first $k - 1$ timesteps. Multistep integrators are not economical of memory because they store the coefficients of the entire interpolating polynomial rather than just the present phase-space position.

### 3.4.4 Multivalue integrators

By differentiating the equations of motion $\dot{\mathbf{w}} = \mathbf{f}(\mathbf{w})$ with respect to time, we obtain an expression for $\ddot{\mathbf{w}}$, which involves second derivatives of the potential, $\partial^2\Phi/\partial q_i \partial q_j$. If our Poisson solver delivers reliable values for these second derivatives, it can be advantageous to use $\ddot{\mathbf{w}}$ or even higher time derivatives of $\mathbf{w}$ to determine the interpolating polynomial $\mathbf{w}^{\rm poly}(t)$. Algorithms that

employ the second and higher derivatives of $\mathbf{w}$ are called **multivalue integrators**.

In the simplest case we set $\mathbf{w}^{\text{poly}}(t)$ to the $k$th-order polynomial that matches $\mathbf{w}$ and its first $k$ time derivatives at $t_0$; this provides $k+1$ constraints for the $k+1$ polynomial coefficients and corresponds to predicting $\mathbf{w}(t)$ by its Taylor series expansion around $t_0$. A more satisfactory approach is to determine $\mathbf{w}^{\text{poly}}(t)$ from the values taken by $\mathbf{w}$, $\dot{\mathbf{w}}$, $\ddot{\mathbf{w}}$, etc., at both $t_0$ and $t_1$. Specifically, for even $k$ only, we make $\mathbf{w}^{\text{poly}}(t)$ the $k$th-order polynomial that matches $\mathbf{w}$ at $t_0$ and its first $\frac{1}{2}k$ time derivatives at both $t_0$ and $t_1$—once again this provides $1 + 2 \times \frac{1}{2}k = k + 1$ constraints and hence determines the $k+1$ coefficients of the interpolating polynomial. The first few integrators of this type are

$$\mathbf{w}_1 = \mathbf{w}_0 + \begin{cases} \frac{1}{2}h(\dot{\mathbf{w}}_0 + \dot{\mathbf{w}}_1) & (k=2) \\ \frac{1}{2}h(\dot{\mathbf{w}}_0 + \dot{\mathbf{w}}_1) + \frac{1}{12}h^2(\ddot{\mathbf{w}}_0 - \ddot{\mathbf{w}}_1) & (k=4) \\ \frac{1}{2}h(\dot{\mathbf{w}}_0 + \dot{\mathbf{w}}_1) + \frac{1}{10}h^2(\ddot{\mathbf{w}}_0 - \ddot{\mathbf{w}}_1) & \\ \qquad + \frac{1}{120}h^3(\dddot{\mathbf{w}}_0 + \dddot{\mathbf{w}}_1) & (k=6). \end{cases} \tag{3.171}$$

Like the Adams–Moulton integrator, all of these integrators are implicit, and in fact the first of these formulae is the same as the second-order Adams–Moulton integrator in equation (3.170). Because these integrators employ information from only $t_0$ and $t_1$, there are two significant simplifications compared to multistep integrators: no separate startup procedure is needed, and the formulae look the same even if the timestep is variable.

Multivalue integrators are sometimes called **Obreshkov** (or Obrechkoff) or **Hermite** integrators, the latter name arising because they are based on Hermite interpolation, which finds a polynomial that fits specified values of a function and its derivatives (Butcher 1987).

Makino & Aarseth (1992) and Makino (2001) recommend a fourth-order multivalue predictor-corrector integrator for star-cluster simulations. Their predictor is a single-step, second-order multivalue integrator, that is, a Taylor series including terms of order $h^2$. Writing $\mathrm{d}\mathbf{v}/\mathrm{d}t = \mathbf{g}$, where $\mathbf{g}$ is the gravitational field, their predicted velocity is

$$\mathbf{v}_{\mathrm{p},1} = \mathbf{v}_0 + h\mathbf{g}_0 + \tfrac{1}{2}h^2\dot{\mathbf{g}}_0. \tag{3.172a}$$

The predicted position is obtained by analytically integrating the interpolating polynomial for $\mathbf{v}$,

$$\mathbf{x}_{\mathrm{p},1} = \mathbf{x}_0 + h\mathbf{v}_0 + \tfrac{1}{2}h^2\mathbf{g}_0 + \tfrac{1}{6}h^3\dot{\mathbf{g}}_0. \tag{3.172b}$$

The predicted position and velocity are used to compute the gravitational field and its time derivative at time $t_1$, $\mathbf{g}_1$ and $\dot{\mathbf{g}}_1$. These are used to correct the velocity using the fourth-order formula (3.171):

$$\mathbf{v}_1 = \mathbf{v}_0 + \tfrac{1}{2}h(\mathbf{g}_0 + \mathbf{g}_1) + \tfrac{1}{12}h^2(\dot{\mathbf{g}}_0 - \dot{\mathbf{g}}_1); \tag{3.172c}$$

in words, $\mathbf{v}_1$ is determined by the fourth-order interpolating polynomial $\mathbf{v}^{\rm poly}(t)$ that satisfies the five constraints $\mathbf{v}^{\rm poly}(t_0) = \mathbf{v}_0$, $\dot{\mathbf{v}}^{\rm poly}(t_i) = \mathbf{g}_i$, $\ddot{\mathbf{v}}^{\rm poly}(t_i) = \dot{\mathbf{g}}_i$ for $i = 0, 1$.

To compute the corrected position, the most accurate procedure is to integrate analytically the interpolating polynomial for $\mathbf{v}$, which yields:

$$\mathbf{x}_1 = \mathbf{x}_0 + h\mathbf{v}_0 + \tfrac{1}{20}h^2(7\mathbf{g}_0 + 3\mathbf{g}_1) + \tfrac{1}{60}h^3(3\dot{\mathbf{g}}_0 - 2\dot{\mathbf{g}}_1). \tag{3.172d}$$

The performance of this integrator, often simply called the Hermite integrator, is illustrated in Figure 3.21.

### 3.4.5 Adaptive timesteps

Except for the simplest problems, any integrator should have an **adaptive timestep**, that is, an automatic procedure that continually adjusts the timestep to achieve some target level of accuracy. Choosing the right timestep is one of the most challenging tasks in designing a numerical integration scheme. Many sophisticated procedures are described in publicly available integration packages and numerical analysis textbooks. Here we outline a simple approach.

Let us assume that our goal is that the error in $\mathbf{w}$ after some short time $\tau$ should be less than $\epsilon|\mathbf{w}_0|$, where $\epsilon \ll 1$ and $\mathbf{w}_0$ is some reference phase-space position. We first move from $\mathbf{w}$ to $\mathbf{w}_2$ by taking two timesteps of length $h \ll \tau$. Then we return to $\mathbf{w}$ and take one step of length $2h$ to reach $\mathbf{w}_1$. Suppose that the correct position after an interval $2h$ is $\mathbf{w}'$, and that our integrator has order $k$. Then the errors in $\mathbf{w}_1$ and $\mathbf{w}_2$ may be written

$$\mathbf{w}_1 - \mathbf{w}' \simeq (2h)^{k+1}\mathbf{E} \quad ; \quad \mathbf{w}_2 - \mathbf{w}' \simeq 2h^{k+1}\mathbf{E}, \tag{3.173}$$

where $\mathbf{E}$ is an unknown error vector. Subtracting these equations to eliminate $\mathbf{w}'$, we find $\mathbf{E} \simeq (\mathbf{w}_1 - \mathbf{w}_2)/[2(2^k - 1)h^{k+1}]$. Now if we advance for a time $\tau$, using $n \equiv \tau/h'$ timesteps of length $h'$, the error will be

$$\boldsymbol{\Delta} = nh'^{k+1}\mathbf{E} = (\mathbf{w}_1 - \mathbf{w}_2)\frac{\tau h'^k}{2(2^k - 1)h^{k+1}}. \tag{3.174}$$

Our goal that $|\boldsymbol{\Delta}| \lesssim \epsilon|\mathbf{w}_0|$ will be satisfied if

$$h' < h_{\rm max} \equiv \left(2(2^k - 1)\frac{h}{\tau}\frac{\epsilon|\mathbf{w}_0|}{|\mathbf{w}_1 - \mathbf{w}_2|}\right)^{1/k} h. \tag{3.175}$$

If we are using a predictor-corrector scheme, a similar analysis can be used to deduce $h_{\rm max}$ from the difference of the phase-space positions returned by the predictor and the corrector, without repeating the entire predictor-corrector sequence.

### 3.4.6 Individual timesteps

The density in many stellar systems varies by several orders of magnitude between the center and the outer parts, and as a result the crossing time of orbits near the center is much smaller than the crossing time in the outer envelope. For example, in a typical globular cluster the crossing time at the center is $\lesssim 1\,\mathrm{Myr}$, while the crossing time near the tidal radius is $\sim 100\,\mathrm{Myr}$. Consequently, the timestep that can be safely used to integrate the orbits of stars is much smaller at the center than the edge. It is extremely inefficient to integrate *all* of the cluster stars with the shortest timestep needed for *any* star, so integrators must allow individual timesteps for each star.

If the integrator employs an interpolating polynomial, the introduction of individual timesteps is in principle fairly straightforward. To advance a given particle, one uses the most recent interpolating polynomials of all the other particles to predict their locations at whatever times the integrator requires, and then evaluates the forces between the given particle and the other particles.

This procedure makes sense if the Poisson solver uses direct summation (§2.9.1). However, with other Poisson solvers there is a much more efficient approach. Suppose, for example, that we are using a tree code (§2.9.2). Then before a single force can be evaluated, *all* particles have to be sorted into a tree. Once that has been done, it is comparatively inexpensive to evaluate large numbers of forces; hence to minimize the computational work done by the Poisson solver, it is important to evaluate the forces on many particles simultaneously. A **block timestep** scheme makes this possible whilst allowing different timesteps for different particles, by quantizing the timesteps. We now describe how one version of this scheme works with the leapfrog integrator.

We assign each particle to one of $K+1$ classes, such that particles in class $k$ are to be advanced with timestep $h_k \equiv 2^k h$ for $k = 0, 1, 2, \ldots, K$. Thus $h$ is the shortest timestep (class 0) and $2^K h$ is the longest (class $K$). The Poisson solver is used to evaluate the gravitational field at the initial time $t_0$, and each particle is kicked by the impulse $-\frac{1}{2}h_k\boldsymbol{\nabla}\Phi$, corresponding to the first part of the kick-drift-kick leapfrog step (3.166b). In Figure 3.22 the filled semicircles on the left edge of the diagram symbolize these kicks; they are larger at the top of the diagram to indicate that the strength of the kicks increases as $2^k$. Then every particle is drifted through time $h$, and the Poisson solver is used only to find the forces on the particles in class 0, so these particles can be kicked by $-h\boldsymbol{\nabla}\Phi$, which is the sum of the kicks at the end of their first leapfrog step and the start of their second.

Next we drift all particles through $h$ a second time, and use the Poisson solver to find the forces on the particles in both class 0 and class 1. The particles of class 0 are kicked by $-h\boldsymbol{\nabla}\Phi$, and the particles of class 1 are kicked by $-h_1\boldsymbol{\nabla}\Phi = -2h\boldsymbol{\nabla}\Phi$. After an interval $3h$ the particles in class 0 are kicked, after $4h$ the particles in classes 0, 1 and 2 are kicked, etc. This

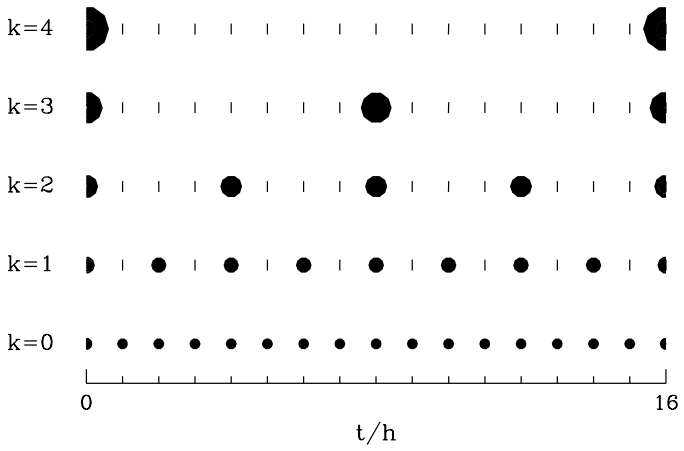


**Figure 3.22** Schematic of the block timestep scheme, for a system with 5 classes of particles, having timestep $h$ (class $k = 0$), $2h, \ldots, 16h$ (class $k = K = 4$). The particles are integrated for a total time of $16h$. Each filled circle or half-circle marks the time at which particles in a given class are kicked. Each vertical bar marks a time at which particles in a class are paused in their drift step, without being kicked, in order to calculate their contribution to the kick given to particles in lower classes. The kicks at the start and end of the integration, $t = 0$ and $t = 16h$, are half as strong as the other kicks, and so are denoted by half-circles.

process continues until all particles are due for a kick, after a time $h_{\rm K} = 2^K h$. The final kick for particles in class $k$ is $-\frac{1}{2}h_k\boldsymbol{\nabla}\Phi$, which completes $2^{K-k}$ leapfrog steps for each particle. At this point it is prudent to reconsider how the particles are assigned to classes in case some need smaller or larger timesteps.

A slightly different block timestep scheme works well with a particle-mesh Poisson solver (§2.9.3) when parts of the computational domain are covered by finer meshes than others, with each level of refinement being by a factor of two in the number of mesh points per unit length (Knebe, Green, & Binney 2001). Then particles are assigned timesteps according to the fineness of the mesh they are in: particles in the finest mesh have timestep $\Delta t = h$, while particles in the next coarser mesh have $\Delta t = 2h$, and so on. Particles on the finest mesh are drifted through time $\frac{1}{2}h$ before the density is determined on this mesh, and the Poisson solver is invoked to determine the forces on this mesh. Then the particles on this mesh are kicked through time $h$ and drifted through time $\frac{1}{2}h$. Then the same drift-kick-drift sequence is used to advance particles on the next coarser mesh through time $2h$. Now these particles are ahead in time of the particles on the finest mesh. This situation is remedied by again advancing the particles on the finest mesh by $h$ with the drift-kick-drift sequence. Once the particles on the two finest

meshes have been advanced through time $2h$, we are ready to advance by $\Delta t = 4h$ the particles that are the next coarser mesh, followed by a repeat of the operations that were used to advance the particles on the two finest meshes by $2h$. The key point about this algorithm is that at each level $k$, particles are first advanced ahead of particles on the next coarser mesh, and then the latter particles jump ahead of the particles on level $k$ so the next time the particles on level $k$ are advanced, they are catching up with the particles of the coarser mesh. Errors arising from moving particles in a gravitational field from the surroundings that is out-of-date are substantially canceled by errors arising from moving particles in an ambient field that has run ahead of itself.

### 3.4.7 Regularization

In any simulation of a star cluster, sooner or later two particles will suffer an encounter having a very small impact parameter. In the limiting case in which the impact parameter is exactly zero (a **collision orbit**), the equation of motion for the distance $r$ between the two particles is (eq. D.33)

$$\ddot{r} = -GM/r^2, \tag{3.176}$$

where $M$ is the sum of the masses of the two particles. This equation is singular at $r = 0$, and a conscientious integrator will attempt to deal with the singularity by taking smaller and smaller timesteps as $r$ diminishes, thereby bringing the entire N-body integration grinding to a halt. Even in a near-collision orbit, the integration through pericenter will be painfully slow. This problem is circumvented by transforming to a coordinate system in which the two-body problem has no singularity—this procedure is called **regularization** (Stiefel & Schiefele 1971; Mikkola 1997; Heggie & Hut 2003; Aarseth 2003). Standard integrators can then be used to solve the equations of motion in the regularized coordinates.

**(a) Burdet–Heggie regularization** The simplest approach to regularization is time transformation. We write the equations of motion for the two-body problem as

$$\ddot{\mathbf{r}} = -GM\frac{\mathbf{r}}{r^3} + \mathbf{g}, \tag{3.177}$$

where $\mathbf{g}$ is the gravitational field from the other $N-2$ bodies in the simulation, and change to a fictitious time $\tau$ that is defined by

$$\mathrm{d}t = r\,\mathrm{d}\tau. \tag{3.178}$$

Denoting derivatives with respect to $\tau$ by a prime we find

$$\dot{\mathbf{r}} = \frac{\mathrm{d}\tau}{\mathrm{d}t}\frac{\mathrm{d}\mathbf{r}}{\mathrm{d}\tau} = \frac{1}{r}\mathbf{r}' \quad ; \quad \ddot{\mathbf{r}} = \frac{\mathrm{d}\tau}{\mathrm{d}t}\frac{\mathrm{d}}{\mathrm{d}\tau}\frac{1}{r}\mathbf{r}' = \frac{1}{r^2}\mathbf{r}'' - \frac{r'}{r^3}\mathbf{r}'. \tag{3.179}$$

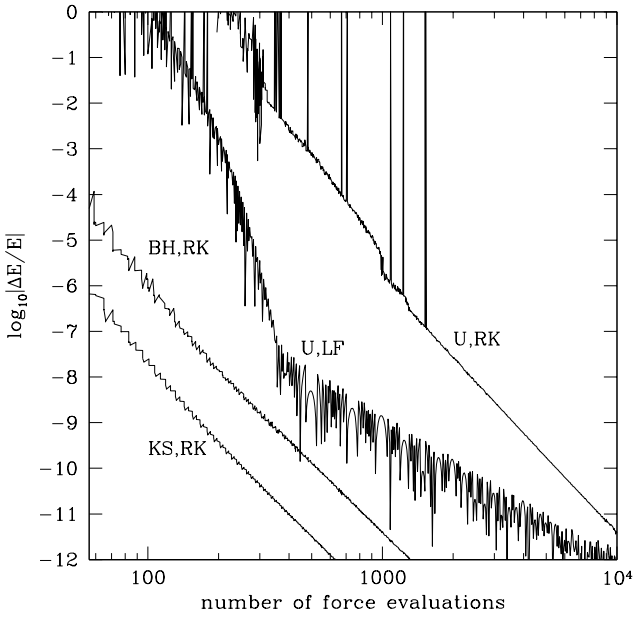


**Figure 3.23** Fractional energy error from integrating one pericenter passage of a highly eccentric orbit in a Keplerian potential, as a function of the number of force evaluations. The orbit has semi-major axis $a = 1$ and eccentricity $e = 0.99$, and is integrated from $r = 1, \dot{r} < 0$ to $r = 1, \dot{r} > 0$. Curves labeled by "RK" are followed using a fourth-order Runge–Kutta integrator (3.168) with adaptive timestep control as described by Press et al. (1986). The curve labeled "U" for "unregularized" is integrated in Cartesian coordinates, the curve "BH" uses Burdet–Heggie regularization, and the curve "KS" uses Kustaanheimo–Stiefel regularization. The curve labeled "U,LF" is followed in Cartesian coordinates using a leapfrog integrator with timestep proportional to radius (eq. 3.167). The horizontal axis is the number of force evaluations used in the integration.

Substituting these results into the equation of motion, we obtain

$$\mathbf{r}'' = \frac{r'}{r}\mathbf{r}' - GM\frac{\mathbf{r}}{r} + r^2\mathbf{g}. \tag{3.180}$$

The eccentricity vector $\mathbf{e}$ (eq. 4 of Box 3.2) helps us to simplify this equation. We have

$$\begin{aligned} \mathbf{e} &= \mathbf{v} \times (\mathbf{r} \times \mathbf{v}) - GM\hat{\mathbf{e}}_r \\ &= |\mathbf{r}'|^2 \frac{\mathbf{r}}{r^2} - \frac{r'}{r}\mathbf{r}' - GM\frac{\mathbf{r}}{r}, \end{aligned} \tag{3.181}$$

where we have used $\mathbf{v} = \dot{\mathbf{r}} = \mathbf{r}'/r$ and the vector identity (B.9). Thus equation (3.180) can be written

$$\mathbf{r}'' = |\mathbf{r}'|^2 \frac{\mathbf{r}}{r^2} - 2GM\frac{\mathbf{r}}{r} - \mathbf{e} + r^2\mathbf{g}. \tag{3.182}$$

The energy of the two-body orbit is

$$E_2 = \tfrac{1}{2}v^2 - \frac{GM}{r} = \frac{|\mathbf{r}'|^2}{2r^2} - \frac{GM}{r}, \tag{3.183}$$

so we arrive at the regularized equation of motion

$$\mathbf{r}'' - 2E_2\mathbf{r} = -\mathbf{e} + r^2\mathbf{g}, \tag{3.184}$$

in which the singularity at the origin has disappeared. This must be supplemented by equations for the rates of change of $E_2$, $\mathbf{e}$, and $t$ with fictitious time $\tau$,

$$E_2' = \mathbf{g}\cdot\mathbf{r}' \quad ; \quad \mathbf{e}' = 2\mathbf{r}(\mathbf{r}'\cdot\mathbf{g}) - \mathbf{r}'(\mathbf{r}\cdot\mathbf{g}) - \mathbf{g}(\mathbf{r}\cdot\mathbf{r}') \quad ; \quad t' = r. \tag{3.185}$$

When the external field $\mathbf{g}$ vanishes, the energy $E_2$ and eccentricity vector $\mathbf{e}$ are constants, the equation of motion (3.184) is that of a harmonic oscillator that is subject to a constant force $-\mathbf{e}$, and the fictitious time $\tau$ is proportional to the eccentric anomaly (Problem 3.29).

Figure 3.23 shows the fractional energy error that arises in the integration of one pericenter passage of an orbit in a Kepler potential with eccentricity $e = 0.99$. The error is plotted as a function of the number of force evaluations; this is the correct economic model if force evaluations dominate the computational cost, as is true for N-body integrations with $N \gg 1$. Note that even with $\gtrsim 1000$ force evaluations per orbit, a fourth-order Runge–Kutta integrator with adaptive timestep is sometimes unable to follow the orbit. Using the same integrator, Burdet–Heggie regularization reduces the energy error by almost five orders of magnitude.

This figure also shows the energy error that arises when integrating the same orbit using leapfrog with adaptive timestep (eq. 3.167) in unregularized coordinates. Even though leapfrog is only second-order, it achieves an accuracy that substantially exceeds that of the fourth-order Runge–Kutta integrator in unregularized coordinates, and approaches the accuracy of Burdet–Heggie regularization. Thus a time-symmetric leapfrog integrator provides much of the advantage of regularization without coordinate or time transformations.

**(b) Kustaanheimo–Stiefel (KS) regularization** An alternative regularization procedure, which involves the transformation of the coordinates in addition to time, can be derived using the symmetry group of the Kepler problem, the theory of quaternions and spinors, or several other methods (Stiefel & Schiefele 1971; Yoshida 1982; Heggie & Hut 2003). Once again we use the fictitious time $\tau$ defined by equation (3.178). We also define a four-vector $\mathbf{u} = (u_1, u_2, u_3, u_4)$ that is related to the position $\mathbf{r} = (x, y, z)$ by

$$\begin{aligned} u_1^2 &= \tfrac{1}{2}(x+r)\cos^2\psi \qquad & u_2 &= \frac{yu_1 + zu_4}{x+r} \\ u_4^2 &= \tfrac{1}{2}(x+r)\sin^2\psi \qquad & u_3 &= \frac{zu_1 - yu_4}{x+r}, \end{aligned} \tag{3.186}$$

where $\psi$ is an arbitrary parameter. The inverse relations are

$$x = u_1^2 - u_2^2 - u_3^2 + u_4^2 \ ; \ y = 2(u_1u_2 - u_3u_4) \ ; \ z = 2(u_1u_3 + u_2u_4). \quad (3.187)$$

Note that $r = u_1^2 + u_2^2 + u_3^2 + u_4^2$. Let $\Phi_{\rm e}$ be the potential that generates the external field $\mathbf{g} = -\boldsymbol{\nabla}\Phi_{\rm e}$. Then in terms of the new variables the equation of motion (3.177) reads

$$\begin{aligned} \mathbf{u}'' - \tfrac{1}{2}E\mathbf{u} &= -\tfrac{1}{4}\frac{\partial}{\partial \mathbf{u}}\left(|\mathbf{u}|^2\Phi_{\rm e}\right), \\ E = \tfrac{1}{2}v^2 - \frac{GM}{r} + \Phi_{\rm e} &= 2\frac{|\mathbf{u}'|^2}{|\mathbf{u}|^2} - \frac{GM}{|\mathbf{u}|^2} + \Phi_{\rm e}, \\ E' = |\mathbf{u}|^2\frac{\partial \Phi_{\rm e}}{\partial t} \quad &; \quad t' = |\mathbf{u}|^2, \end{aligned} \quad (3.188)$$

When the external force vanishes, the first of equations (3.188) is the equation of motion for a four-dimensional harmonic oscillator.

Figure 3.23 shows the fractional energy error that arises in the integration of an orbit with eccentricity $e = 0.99$ using KS regularization. Using the same integrator, the energy error is more than an order of magnitude smaller than the error using Burdet–Heggie regularization.

## 3.5 Angle-action variables

In §3.1 we introduced the concept of an integral of motion and we saw that every spherical potential admitted at least four integrals $I_i$, namely, the Hamiltonian and the three components of angular momentum. Later we found that orbits in flattened axisymmetric potentials frequently admit three integrals, the classical integrals $H$ and $p_\phi$, and the non-classical third integral. Finally in §3.3 we found that many orbits in planar non-axisymmetric potentials admitted a non-classical integral in addition to the Hamiltonian.

In this section we explore the advantages of using integrals as coordinates for phase space. Since elementary Newtonian or Lagrangian mechanics restricts our choice of coordinates to ones that are rarely integrals, we work in the more general framework of Hamiltonian mechanics (Appendix D). For definiteness, we shall assume that there are three independent coordinates (so phase space is six-dimensional) and that we have three analytic isolating integrals $I_i(\mathbf{x}, \mathbf{v})$. We shall focus on a particular set of canonical coordinates, called **angle-action** variables; the three momenta are integrals, called "actions," and the conjugate coordinates are called "angles." An orbit fortunate enough to possess angle-action variables is called a **regular orbit**.

We start with a number of general results that apply to any system of angle-action variables. Then in a series of subsections we obtain explicit