

Fundamentals of Simulation Methods – Exam Sheet

Janosh Riebesell, February 2016

1 Issues of floating point math

- common problems: **overflow** and **round-off errors**
- **two's complement**: *most significant bit* (MSB) (a.k.a. sign bit s) flags negative numbers
- floating point representations have a **base** β ($\beta = 2$ in IEEE standard), **precision** p (a.k.a. *mantissa*), and **exponent** e
 - representation not unique, shifting exponent *and* mantissa can give same number multiple times; solution: *normalization*, i.e. leading digit before the point always 1
 - number in exponent E stored with *bias* b (hence E always positive as opposed to two's complement), e.g. $e = E - 127$ for floats
- in general, value f of IEEE float given by $f = (-1)^s (1 + \frac{M}{2^p}) \cdot 2^{E-b}$
- **machine precision**: smallest increment in mantissa, $\epsilon_m = \frac{1}{2^p}$
 - pitfall: $a + b = a$ even for $b \neq 0$ if $|b| < \epsilon_m |a|$

2 Integration of ordinary differential equations

- we seek solution $\mathbf{y}(t)$ to eq. of the form $\frac{d\mathbf{y}}{dt} = \mathbf{f}(\mathbf{y}, t)$ subject to boundary condition, most commonly *initial value problem* (IVP)
- forward or **explicit Euler method**: $\mathbf{y}_{n+1} = \mathbf{y}_n + \mathbf{f}(\mathbf{y}_n)\Delta t + \mathcal{O}_s(\Delta t^2)$
 - simplest approach, but only first-order accurate $\mathcal{O}_s(\Delta t^2)$ and unstable for large stepsize Δt
 - explicit because r.h.s. depends only on known quantities
- backward or **implicit Euler method**: $\mathbf{y}_{n+1} = \mathbf{y}_n + \mathbf{f}(\mathbf{y}_{n+1})\Delta t$
 - excellent stability, hence useful for *stiff eqs.*, but still inaccurate for large Δt
 - often involves root-finding of nonlinear eq., numerically expensive
- **implicit midpoint rule**: $\mathbf{y}_{n+1} = \mathbf{y}_n + \mathbf{f}\left(\frac{\mathbf{y}_n + \mathbf{y}_{n+1}}{2}\right)\Delta t + \mathcal{O}_s(\Delta t^3)$
 - second-order accurate but implicit so difficult in practice
- **Runge-Kutta methods**: derive from exact solution $\mathbf{y}_{n+1} = \mathbf{y}_n + \int_{t_n}^{t_{n+1}} \mathbf{f}(\mathbf{y}(t)) dt$ by approximating integral with trapezoidal rule:

$$\begin{aligned} \mathbf{k}_1 &= \mathbf{f}(\mathbf{y}_n, t_n), \quad \mathbf{k}_2 = \mathbf{f}\left(\mathbf{y}_n + \mathbf{k}_1 \Delta t, t_n + \frac{\Delta t}{2}\right), \\ \mathbf{y}_{n+1} &= \mathbf{y}_n + \frac{1}{2}(\mathbf{k}_1 + \mathbf{k}_2)\Delta t + \mathcal{O}_s(\Delta t^3) \end{aligned} \quad (1)$$

- a.k.a. predictor-corrector scheme, $\mathbf{k}_1$ = predictor, $\mathbf{k}_2$ = corrector
- 2nd-order accurate, higher-order schemes exist, e.g. RK-4:

$$\begin{aligned} \mathbf{k}_1 &= \mathbf{f}(\mathbf{y}_n, t_n), \quad \mathbf{k}_2 = \mathbf{f}\left(\mathbf{y}_n + \mathbf{k}_1 \frac{\Delta t}{2}, t_n + \frac{\Delta t}{2}\right), \\ \mathbf{k}_3 &= \mathbf{f}\left(\mathbf{y}_n + \mathbf{k}_2 \frac{\Delta t}{2}, t_n + \frac{\Delta t}{2}\right), \quad \mathbf{k}_4 = \mathbf{f}(\mathbf{y}_n + \mathbf{k}_3 \Delta t, t_n + \Delta t), \\ \mathbf{y}_{n+1} &= \mathbf{y}_n + \frac{1}{6}(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4)\Delta t + \mathcal{O}_s(\Delta t^5) \end{aligned} \quad (2)$$

- **adaptive step size**: achieve optimum compromise between accuracy, efficiency and stability by e.g. estimating error based on difference between one full step and two half-steps, define local error bounds, increase stepsize if below, decrease if above
- **reduction to 1st order**: bring higher-order diff. eqs. into standard form, e.g. cast $\ddot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$ into $\dot{\mathbf{y}} = \tilde{\mathbf{f}}(\mathbf{y})$ by defining $\mathbf{y} = (\mathbf{x}, \dot{\mathbf{x}})$ and $\tilde{\mathbf{f}} = (\dot{\mathbf{x}}, \mathbf{f}(\mathbf{x}))$, then apply e.g. Runge-Kutta

- **leapfrog**: very simple alternative in above case; define $\mathbf{v} \equiv \dot{\mathbf{x}}$, then

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \mathbf{v}_{n+\frac{1}{2}} \Delta t, \quad \mathbf{v}_{n+\frac{3}{2}} = \mathbf{v}_{n+\frac{1}{2}} + \mathbf{f}(\mathbf{x}_{n+1})\Delta t \quad (3)$$

- 2nd-order accurate and requires only one evaluation per step
- name leapfrog due to interleaved advances of position and velocity
- better than higher-order schemes on some systems due to being symplectic and time-reversible
- **symplectic integrator**: structure-preserving method that observes special properties of Hamiltonian systems, e.g. energy and phase-space volume conservation (*Liouville theorem*)
 - let $\omega: \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$, $\omega(\xi, \eta) = A_{P(\xi \times \eta)}$ give area of parallelogram $P(\xi \times \eta)$ spanned by $\xi, \eta \in \mathbb{R}^d$; then linear map $F: \mathbb{R}^d \rightarrow \mathbb{R}^d$ is symplectic if $\omega(F\xi, F\eta) = \omega(\xi, \eta)$
 - a differentiable map $g: U \rightarrow \mathbb{R}^d$ is symplectic if its Jacobian is symplectic everywhere
 - **Poincaré's theorem**: time evolution generated by a Hamiltonian in phase-space is a symplectic transformation

3 Collisionless particle systems

- central idea: describe large- N -body system as small- N system by grouping real particles into fiducial macro particles
 - for gravity, the e.o.m.s with (numerical) *softening length* ϵ are $\ddot{\mathbf{x}}_i = -\nabla\Phi(\mathbf{x}_i)$, $\Phi(\mathbf{x}) = -G \sum_{j=1}^N \frac{m_j}{\sqrt{(\mathbf{x} - \mathbf{x}_j)^2 + \epsilon^2}}$
 - purpose of ϵ : prevent (highly correlated) bound states (requires $\langle v^2 \rangle \gg \frac{Gm}{\epsilon}$), avoid numerical expense of large angle scattering in singular potential; ϵ introduces smallest resolved length scale
- examples: electrons in plasma, globular star clusters, stars in a galaxy, dark matter (all are also *uncorrelated* to good approx.)

4 Tree algorithms

- more spec. **hierarchical multipole algorithms** enable fast, approximate force calculations for N -body systems (see Section 3)
 - group hierarchy organized in *tree-like* data structure
 - scale with $\mathcal{O}(N \ln N)$, whereas *direct summation* is $\mathcal{O}(N^2)$
- central idea: use multipole expansion for (grav.) potential created by group of distant particles rather than sum up forces individually
 - *monopole*: $M = \sum_{i=1}^N m_i$, *quadrupole*: $Q_{ij} = \sum_{k=1}^N m_k [3(\mathbf{s} - \mathbf{x}_k)_i (\mathbf{s} - \mathbf{x}_k)_j - \delta_{ij} (\mathbf{s} - \mathbf{x}_k)^2]$, where $\mathbf{s}$ points to c.o.m.
 - dipole vanishes when expansion is done relative to c.o.m.
 - expansion accurate if $\frac{r}{d} \leq \theta$ for chosen **opening angle** θ ; r, d radius and distance of particle group

5 The particle-mesh technique

- alternative to tree algorithms, used in MD and astrophysics
- 1. construct density field ρ on suitable auxiliary mesh 2. compute potential Φ on mesh by solving Poisson's eq. 3. calculate force field $\mathbf{F}$ from potential 4. calculate forces $\mathbf{F}_i$ at original particle positions
- 2. mass assignment onto auxiliary mesh via *nearest grid point* (NGP), *clouds in cell* (CIC), or *triangular shaped clouds* (TSC)

- NGP: density jumps discontin. when particle crosses cell boundary, resulting force piece-wise constant; CIC: force piece-wise linear and continuous, but derivative jumps; TSC: both force and its derivative smooth
- 3. solve Poisson's eq. $\Delta\Phi = 4\pi G\rho$ either with Fourier transform (Section 6) or iteratively (Section 7)
- 4. calculate acceleration $\mathbf{a} = -\nabla\Phi$ with **finite differencing**, e.g. $a_x(i, j, k) = -\frac{1}{2h} [\Phi(i+1, j, k) - \Phi(i-1, j, k)] + \mathcal{O}(h^2)$ or with *larger stencil* to reduce truncation error below $\mathcal{O}(h^2)$
- 5. interpolate from mesh to particles: $\mathbf{F}_i(\mathbf{x}_i) = m \sum_{\mathbf{p}} \mathbf{a}_{\mathbf{p}} W(\mathbf{x}_i - \mathbf{x}_{\mathbf{p}})$, where sum runs over auxiliary mesh cells $\mathbf{p} = (i, j, k)$
 - W same assignment kernel as used in density construction, ensures vanishing *self-force* and pairwise antisymmetric forces

6 Force calculation with Fourier transform techniques

- tool for solving certain PDEs, e.g. Poisson's eq. $\Delta\Phi = 4\pi G\rho$; sol. $\Phi(\mathbf{x}) = -G \int \frac{\rho(\mathbf{x}') d\mathbf{x}'}{|\mathbf{x} - \mathbf{x}'|} = \rho \star g$ is convolution of ρ with $g(\mathbf{x}) = -\frac{G}{|\mathbf{x}|}$
- **convolution thm.:** $\mathcal{F}(f \star g) = \mathcal{F}(f) \cdot \mathcal{F}(g) \Rightarrow \Phi = \mathcal{F}^{-1}[\mathcal{F}(\rho) \cdot \mathcal{F}(g)]$
- in practice, to solve this requires assuming *periodic* b.c. in box of size L , gives finite discrete $\mathbf{k}$ -space so we can apply (*discrete*) *fast Fourier transform* (FFT), scales with $\mathcal{O}(N \ln N)$
- for nonperiodic problems: **zero-padding** constructs mesh s.t. sources lives only in one quarter (in 2D), rest of mesh is zeroed-out

7 Iterative solvers and the multigrid technique

- ansatz: discretize Φ on regular N -point mesh with spacing h so Poisson's eq. at cell $i \in \{0, \dots, N-1\}$ is $\frac{\Phi_{i+1} - 2\Phi_i + \Phi_{i-1}}{h^2} = 4\pi G\rho_i$
- above can be cast into set of N linear eqs. $\mathbf{A}\Phi = \mathbf{b}$ with $\mathbf{A}$ stencil matrix, $\mathbf{b} = 4\pi G(\rho_0, \dots, \rho_{N-1})$; analytic. solvable with Gauss elimination, but not feasible for large N (scales $\mathcal{O}(N^3)$)
- use **Jacobi** or **Gauss-Seidel iteration** instead by decomposing $\mathbf{A} = \mathbf{D} - \mathbf{U} - \mathbf{L}$, yields Jac. update formula $\Phi_i^{(n+1)} = \frac{1}{2} (\Phi_{i+1}^{(n)} + \Phi_{i-1}^{(n)} - 4\pi G h^2 \rho_i)$ or GS $\Phi^{(n+1)} = \mathbf{D}^{-1} [\mathbf{U}\Phi^{(n)} + \mathbf{L}\Phi^{(n+1)} + \mathbf{b}]$
 - GS looks like, but is not implicit formula; no values of $\Phi^{(n+1)}$ are needed, because $\mathbf{L}$ has only elements below diagonal
 - GS makes use of updated values immediately, $\approx$ twice as fast as Jacobi, but has to be solved sequentially $\Rightarrow$ not parallelizable
- **red-black ordering**: some update rules allow decomposition of cells into disjoint sets whose update depends only on cells from other set(s), e.g. for Poisson's eq.: chessboard pattern
- convergence of Jac. and GS often '*stalls*'; high- ν errors die quickly, but low- ν , high- λ errors slowly; reason: per step, information travels only one stencil length, convergence needs several sweeps
- solution: **multigrid**; central idea: go to coarser mesh, compute improved initial guess, helps speed up convergence on fine grid
 - because multigrid operates on coarser grid for most of time, it is *much* faster, cheaper than Jac. and GS
 - fine $\rightarrow$ coarse and back requires restriction and prolongation operators; in 1D: restr. with $\mathbf{I}_h^{2h} = [\frac{1}{4}, \frac{1}{2}, \frac{1}{4}]$, prol. with $\mathbf{I}_{2h}^h = [\frac{1}{2}, 1, \frac{1}{2}]$

8 Molecular dynamics simulations

- aim: simulate atoms, molecules, proteins, etc. in *microscopic* detail over relevant timespan to determine mech., electr., etc. properties
- important point: uses Newton's *classical* e.o.m.s, based on approximation of inter- and/or intra-molecular forces, force laws may be empirical or quantum mechanically derived
- intimately connected to stat. mech.: precise microstate unimportant, interested in *ensemble averages* of macro. var. T, P, ρ , etc.
- **ergodic hypothesis**: postulates ensemble avr. = time avr., thus evolve MD sim. to reach equilibrium, then measure time avr. $\bar{A} = \lim_{\tau \rightarrow \infty} \frac{1}{\tau} \int_{t_0}^{t_0+\tau} A(\Gamma(t)) dt$ of macros. quantity $A(\Gamma)$, fct. of microstate Γ
- **finite range interactions**: molecules interact only over short range, e.g. *Lennard-Jones potential* $V(r) = 4\epsilon \left[\frac{\sigma^{12}}{r^{12}} - \frac{\sigma^6}{r^6} \right] \Rightarrow$ introduce cutoff $V_c(r) = 0$ for $r > r_c$ and search grid, ignore particles in cells farther than r_c ; reduces scaling from $\mathcal{O}(N^2)$ to $\mathcal{O}(N)$

9 Basic gas dynamics

- **Euler eqs.:** describe continuity of mass, momentum, energy during dynamics of ideal, i.e. frictionless, gas

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0, \quad \frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v}^2 + P) = 0, \quad \frac{\partial \rho e}{\partial t} + \nabla \cdot [(\rho e + P) \mathbf{v}] = 0 \quad (4)$$

with total energy (per unit mass) $e = u + \frac{\mathbf{v}^2}{2}$, u thermal energy; closed by eq. of state $P = (\gamma - 1)\rho u$ with adiabatic index $\gamma = C_P/C_V$

- **Navier-Stokes eqs.:** describe dynamics of real fluids with internal stress due to *viscosity* (dissipates relative motion into heat)

$$\frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v}^2 + P) = \nabla \Pi, \quad \frac{\partial \rho e}{\partial t} + \nabla \cdot [(\rho e + P) \mathbf{v}] = \nabla \cdot (\Pi \mathbf{v}) \quad (5)$$

(continuity eq. identical to Euler) with viscous stress tensor (material property) $\Pi = \eta [\nabla \mathbf{v} + (\nabla \mathbf{v})^T - \frac{2}{3} (\nabla \cdot \mathbf{v}) \mathbf{1}] + \xi (\nabla \cdot \mathbf{v}) \mathbf{1}$, where η (ξ) scales traceless (diagonal) part, i.e. shear (bulk) viscosity

- **shocks**: hydro-flows can develop shock waves where density, velocity, temperature, specific entropy jump; typical for hyperbolic PDEs; shocks decelerate, compress and heat up fluid

- differential form of Euler eqs. breaks down at discontinuity, (weak) integral formulation remains valid; assuming rest frame of shock ($v_s = 0$) and piece-wise constant states left/right leads to Rankine-Hugoniot jump cndts.

$$\rho v = \rho' v', \quad \rho v^2 + P = \rho' v'^2 + P', \quad (\rho e + P)v = (\rho' e' + P')v' \quad (6)$$

- presence of shock requires Mach number $\mathcal{M} = \frac{v}{c_s} > 1$, i.e. pre-shock gas streams supersonically into discontinuity, $c_s^2 = \frac{\gamma P}{\rho}$
- **fluid instabilities**: start with two phases $(\rho_1, v_1), (\rho_2, v_2)$, gravity $\mathbf{g}$ in $-z$ -direction, consider single pert. Fourier mode $\xi = \tilde{\xi} e^{i(kx - \omega t)}$
 - **Rayleigh-Taylor**: $v_1 = v_2 = 0 \Rightarrow$ disp. rel. $\omega^2 = \frac{(\rho_2 - \rho_1)kg}{\rho_1 + \rho_2}$ (phase 1 on top); unstable solutions with $\text{Im}(\omega) > 0$ exist for $\rho_1 > \rho_2$, i.e. heavy fluid on top of light one
 - **Kelvin-Helmholtz**: $\mathbf{g} = 0$, i.e. pure shear flow $\Rightarrow \omega_{1,2} = k \frac{(\rho_1 v_1 + \rho_2 v_2)}{\rho_1 + \rho_2} \pm ik \frac{\sqrt{\rho_1 \rho_2}}{\rho_1 + \rho_2} |v_1 - v_2|$; $\text{Im}(\omega) > 0$ always if $v_1 \neq v_2$, i.e. small perturbation at phase interface will grow rapidly into characteristic KH vortices (fluid turbulences)

10 Eulerian hydrodynamics

- important linear and homogeneous PDEs: *Laplace* $\Delta u = 0$, *heat conduction* $\partial_t u = \lambda^2 \Delta u$, *wave propagation* $\partial_t^2 u = c^2 \Delta u$

- classify 2nd-order linear PDEs $a\partial_x^2 u + b\partial_x\partial_y u + c\partial_y^2 u + d\partial_x u + e\partial_y u + fu = g$ according to $D = b^2 - 4ac$: $D < 0$ elliptic (static problems without time dependence), $D = 0$ parabolic (slowly changing processes, e.g. diffusion), $D > 0$ hyperbolic (dynamical processes that can develop discontinuities)
- **Riemann problem**: IVP for hyperbolic system of two (piece-wise constant) phases characterized at $t = 0$ by state vectors $\mathbf{U}_i = (\rho_i, P_i, \mathbf{v}_i)$, $i \in \{R, L\}$, task: evolve system for $t > 0$
 - ▶ (always present) *contact wave*: boundary between phases
 - ▶ may be sandwiched between *shock* and/or *rarefaction waves*
 - ▶ waves propagate with constant speed, fluid properties $(\rho^*, P^*, \mathbf{v}^*)$ at $x = 0$ constant in time for $t > 0$
 - ▶ for $\mathbf{v}_L = \mathbf{v}_R = 0$, Riemann problem simplifies to *Sod shock tube*
- **Metropolis-Hastings algorithm**: 1. from current state x , propose new state x' with proposal probability $q(x \rightarrow x')$ 2. accept proposed move with probability given by *Hastings ratio* $r = \min\left(1, \frac{p(x')q(x \rightarrow x)}{p(x)q(x \rightarrow x')}\right)$ 3. otherwise reject x' and add x to Markov chain (again)
 - ▶ fulfills *detailed balance*: $p_{\text{eq}}(x)W_f(x \rightarrow x') \stackrel{\check}{=} p_{\text{eq}}(x')W_f(x' \rightarrow x)$

11 Smoothed particle hydrodynamics

- (versatile mesh-free) technique to approximate fluid continuum dynamics through particles (interpolation points)
- advantages: 1. provides automatically adaptive spatial resolution and density 2. ensures conservation of energy, (angular) momentum, mass, entropy 3. Galilean invariant and advection-error-free 4. can deal with complicated geometries and large empty regions due to mesh-free nature 5. robust: does not produce negative ρ 's and T 's
- disadvantage: limited accuracy in multi-dim. flows, gives rise to velocity noise, degrades accuracy and leads to slow convergence

12 Finite element methods

- for solving PDEs, good at flexible geometries, odd b.c., allows spatially variable resolution
- central idea: 1. divide domain into smaller regions (*elements*) containing points (*nodes*), possible boundaries: segments (1D), triangles (2D), tetrahedra (3D) 2. choose (linearly indep.) basis fcts. (called *shape fcts.*) to describe solution on each element, e.g. polynomial basis 3. n coefficients in element expansion require n nodes to uniquely specify *reconstruction* inside element

13 Monte Carlo Techniques

- lie at the heart of *stochastic simulations*
- **error** of Monte Carlo integration I_N given by width σ_N of its probability distribution $P_N(I_N)$; always a **Gaussian** according to **central limit theorem** so $\sigma_N = V\sqrt{\frac{\langle f^2 \rangle - \langle f \rangle^2}{N}}$
 - $\propto 1/\sqrt{N}$, irrespective of dimensionality d of integration domain
- **importance sampling**: choose random points preferentially around peaks of function to be integrated
- virtues of **pseudo-random number generators**: repeatability, randomness, speed, portability, long period, insensitivity to seed
- **exact inversion**: method for creating nonuniform PDF $p(y)$ from uniform distr. $q(x) = \frac{1}{b-a}$ over the interval $[a, b]$; uses conservation of probability to equate respective CDFs

$$Q(x) = \int_{-\infty}^x q(x') dx' \stackrel{!}{=} \int_{-\infty}^y p(y') dy' = P(y) \quad (7)$$

- **rejection method**: used to create uninvertible random number distr. $p(y)$ from distr. $f(y) \geq p(y)$ we can produce
 - ▶ 1. generate y from $f(y)$ 2. generate z from uniform distr. with bounds $0 \leq z < f(y)$ 3. if $z \leq p(y)$, return y as sample value 4. else reject trial value for y and start again
- **Markov chain**: discrete sequence of states x_i generated by *Markov process* f which must 1. preserve eq. distr. $p_{\text{eq}}(x)$ of stochastic process 2. starting from x , be able to reach any other state x' (*ergodicity*)