Intro. Lectures 03: Overview of Basic Numerical Methods*

Prof. Steven M. Lund
Physics and Astronomy Department
Facility for Rare Isotope Beams (FRIB)
Michigan State University (MSU)

US Particle Accelerator School (USPAS) Lectures On
"Self-Consistent Simulations of Beam and Plasma Systems"
Steven M. Lund, Jean-Luc Vay, Remi Lehe, and Daniel Winklehner

US Particle Accelerator School Summer Session
Colorado State U, Ft. Collins, CO, 13-17 June, 2016
(Version 20160613)

* Research supported by:
FRIB/MSU, 2014 On via: U.S. Department of Energy Office of Science Cooperative Agreement DE-SC0000661and National Science Foundation Grant No. PHY-1102511
and
LLNL/LBNL, Pre 2014 via: US Dept. of Energy Contract Nos. DE-AC52-07NA27344 and DE-AC02-05CH11231

Detailed Outline
Introductory Lectures on Self-Consistent Simulations

Overview of Basic Numerical Methods

A. Discretization

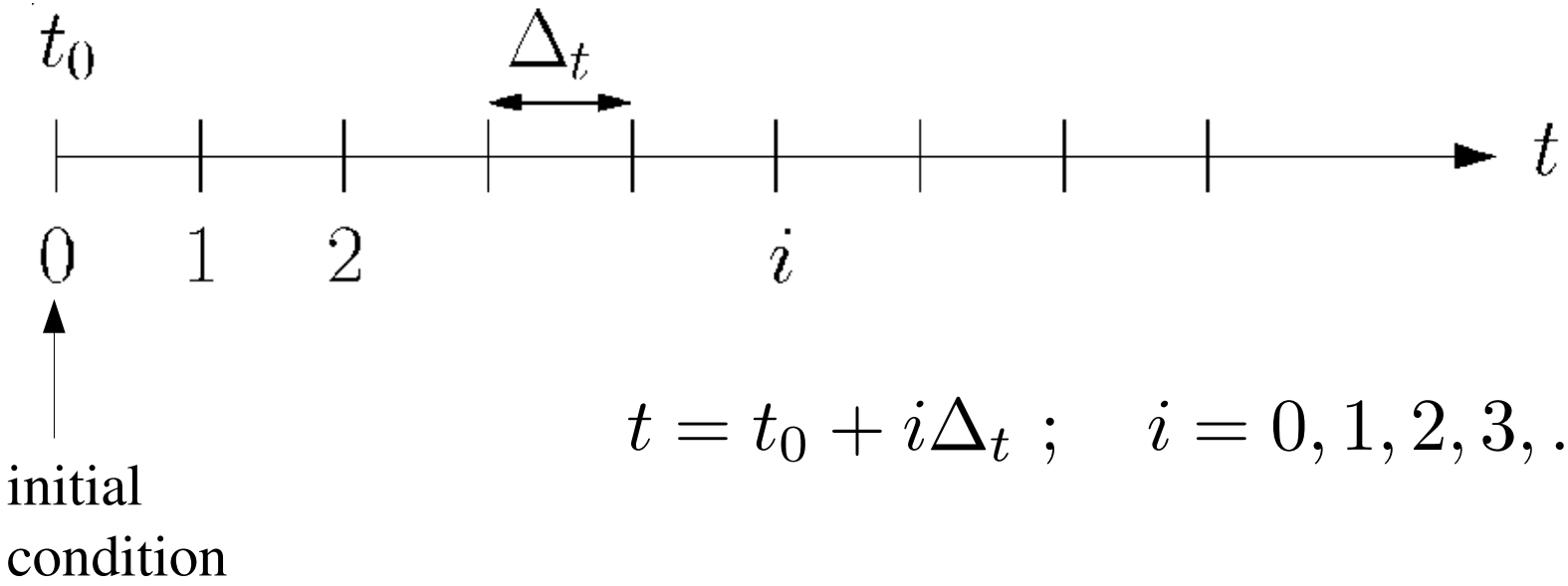
B. Discrete Numerical Operations
- Derivatives
- Quadrature
- Irregular Grids and Axisymmetric Systems

C. Time Advance
- Overview
- Euler and Runge-Kutta Advances
- Solution of Moment Methods

Overview of Basic Numerical Methods
A: Discretizations

General approach is to discretize independent variables in each of the methods and solve for dependent variables which in some cases may be discretized as well

Time (or axial coordinate s)



$$t = t_0 + i\Delta_t \; ; \quad i = 0, 1, 2, 3, ...$$

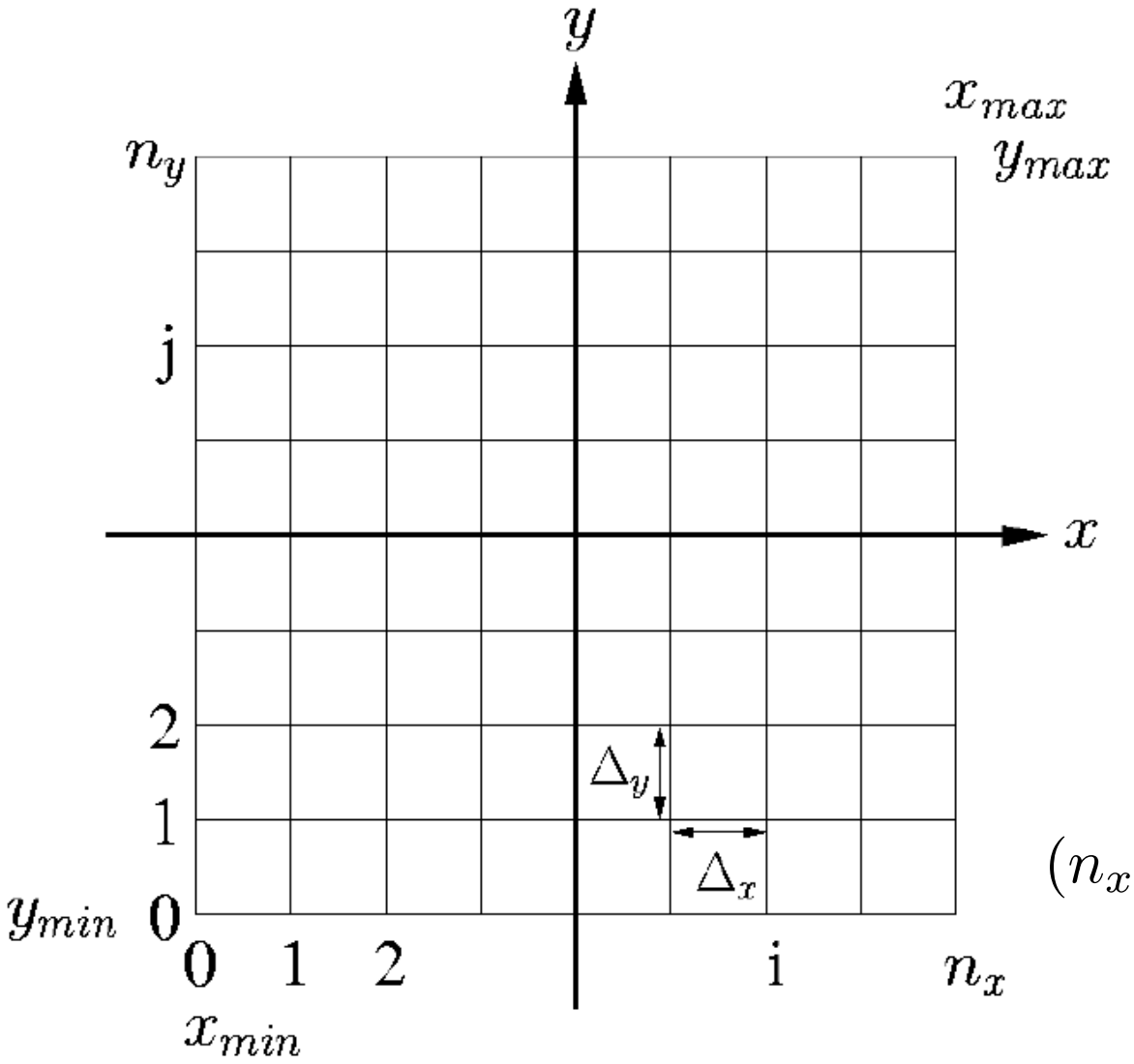
- Nonuniform meshes also possible
  - Add resolution where needed
  - Increases complexity

In applications may apply these descriptions in a variety of ways

- Move a transverse thin slice of a beam, evolve a 3D beam, ...

Transverse Coordinate Discretization

Spatial Coordinates (transverse)



$$x_i = x_{min} + i\Delta_x$$

$$y_j = y_{min} + j\Delta_y$$

$$\Delta_x = (x_{max} - x_{min})/n_x$$

$$\Delta_y = (y_{max} - y_{min})/n_y$$

$$i = 0, 1, 2, ..., n_x$$

$$j = 0, 1, 2, ..., n_y$$

$(n_x + 1)(n_y + 1)$ =Number Mesh Points

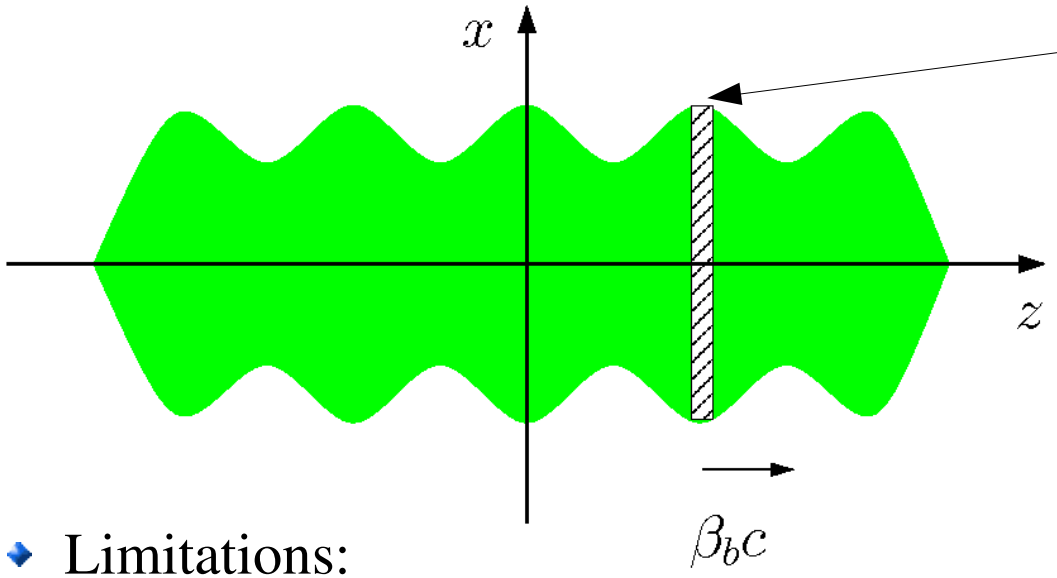
- Analogous for 3D, momentum coordinates (in direct Vlasov simulations), etc.
- Nonuniform meshes possible to add resolution where needed

Transverse Coordinate Discretization – Applications

In applications may apply these discretizations in a variety of ways in distribution type models:

Transverse Slice Simulation:

- Move a transverse thin "slice" of beam along the axial coordinate *s* of a reference particle
  - Many examples/illustrations will be in this context since full 3D straightforward to generalize but is more involved



Thin slice of a long pulse is advanced and the transverse grid moves with the slice

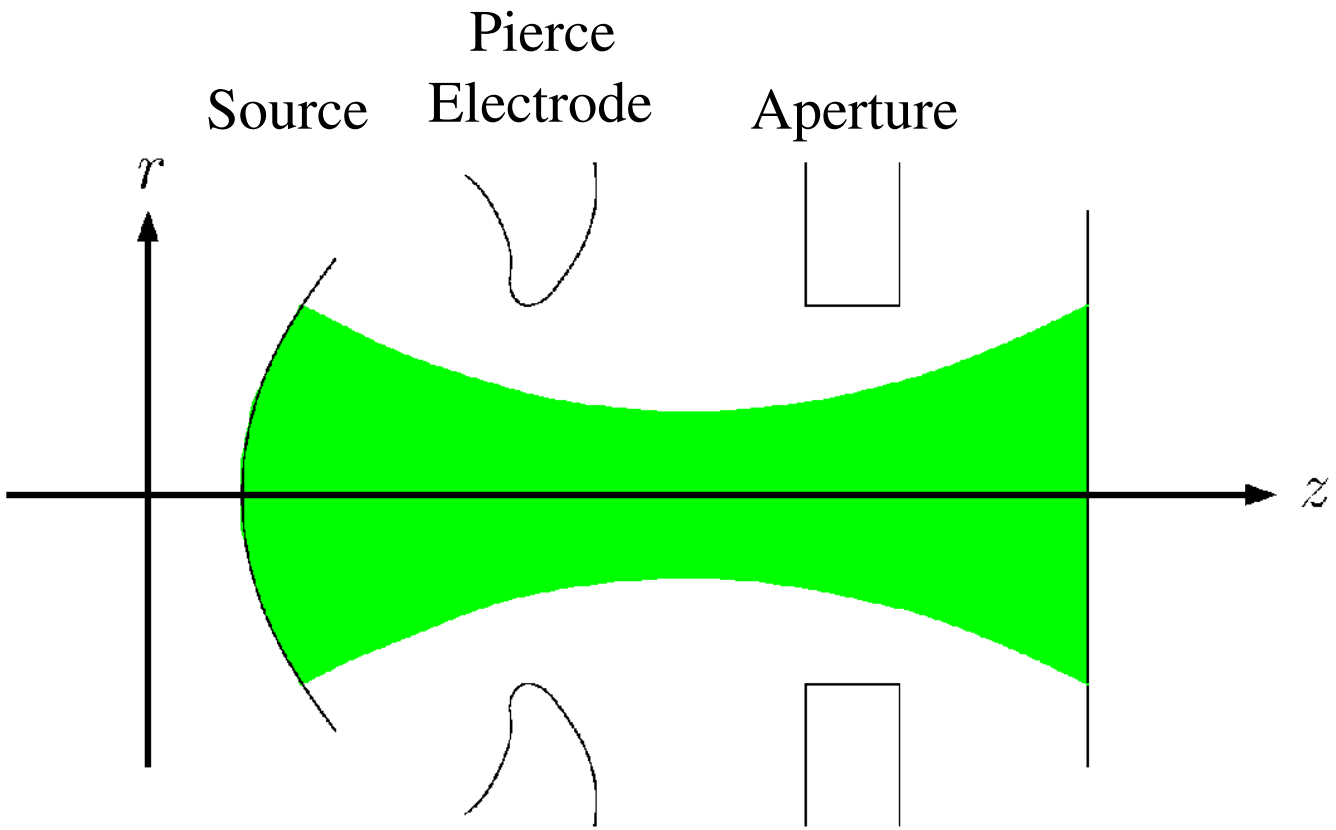
- Limitations:
  - This "unbunched" approximation is not always possible
  - 3D effect can matter, e.g. in short pulses and/or beams ends. Dynamics does not always separate well between transverse and longitudinal effects.

Transverse Coordinate Discretization – Applications (2)

Steady State Simulation:

- Simulate the middle of a long pulse or DC beam where a time stationary beam fills the grid

Example: Mid-Pulse Diode

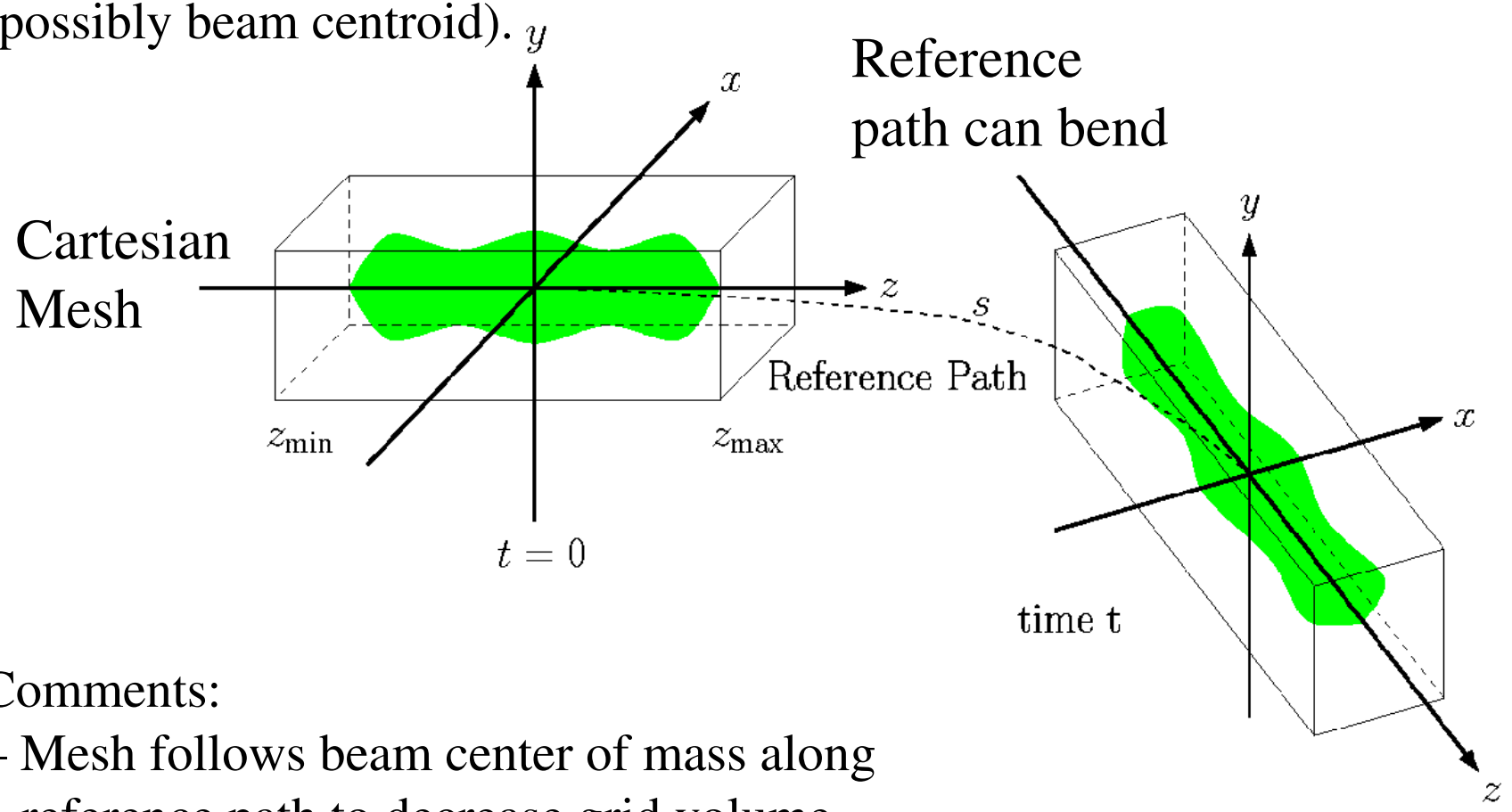


- Mesh is stationary, leading to limitations
  - Pulsed beam always has ends
  - Assumes that the mid-pulse in nearly time-independent in structure

Transverse Coordinate Discretization – Applications (3)

Full 3D Simulation

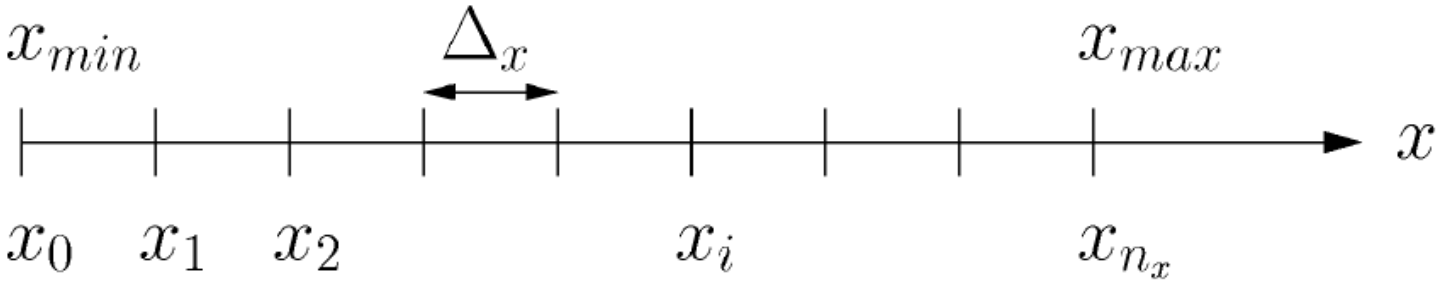
- Simulate a 3D beam with a moving mesh that follows a reference particle (possibly beam centroid).



- Comments:
  - Mesh follows beam center of mass along reference path to decrease grid volume
  - Most realistic level of modeling, but also most numerically intensive
  - Grid can be moved in discretized jumps so that applied fields maintain alignment with the grid

B: Discrete Numerical Operations

Let x represent a spatial coordinate and $f(x)$ some continuous function of x



$$x_i = x_{min} + i\Delta_x; \qquad \Delta_x = (x_{max} - x_{min})/n_x$$

$$i = 0, 1, 2, ..., n_x \qquad (\mathrm{n}_x + 1) = \text{Number Mesh Points}$$

Denote $f_i \equiv f(x_i)$, etc. and Taylor expand one grid point forward and backward about $x = x_i$

$$f_{i+1} = f_i + \left.\frac{\partial f}{\partial x}\right|_i \Delta_x + \frac{1}{2!}\left.\frac{\partial^2 f}{\partial x^2}\right|_i \Delta_x^2 + \frac{1}{3!}\left.\frac{\partial^3 f}{\partial x^3}\right|_i \Delta_x^3 + ...$$

$$f_{i-1} = f_i - \left.\frac{\partial f}{\partial x}\right|_i \Delta_x + \frac{1}{2!}\left.\frac{\partial^2 f}{\partial x^2}\right|_i \Delta_x^2 - \frac{1}{3!}\left.\frac{\partial^3 f}{\partial x^3}\right|_i \Delta_x^3 + ...$$

The same methodology can be applied to other spatial (y, z) coordinates and temporal (t or s) coordinates

Discrete Numerical Operations: Derivatives

Simple, but inaccurate expressions for 1st order derivatives follow immediately from the forward and backward expansions

2 point: (non-centered)

Forward: $$\left.\frac{\partial f}{\partial x}\right|_i = \frac{f_{i+1} - f_i}{\Delta_x} + \mathcal{O}(\Delta_x)$$

Backward: $$\left.\frac{\partial f}{\partial x}\right|_i = \frac{f_i - f_{i-1}}{\Delta_x} + \mathcal{O}(\Delta_x)$$

A more accurate, centered discretization for a 1st order derivative is obtained by subtracting the two expansions.

3 point: (centered)

$$\left.\frac{\partial f}{\partial x}\right|_i = \frac{f_{i+1} - f_{i-1}}{2\Delta_x} + \mathcal{O}(\Delta_x^2)$$

- More accuracy generally will require the use of more function points

Discrete Numerical Operations: Derivatives (2)

The expansions can be relabeled (i -> i+1, etc.) and the resulting set of equations can be manipulated to obtain 5-point and other higher-order forms with higher accuracy:

5 point: (centered)

$$\left.\frac{\partial f}{\partial x}\right|_i = \frac{f_{i-2} - 8f_{i-1} + 8f_{i+1} - f_{i+2}}{12\Delta_x} + \mathcal{O}(\Delta_x^4)$$

- Higher order, and more accurate, forms are possible
  - Rapidly become cumbersome and require more neighboring points

Analogous methods can be employed to obtain discretizations of higher order derivatives. For example,

3 point: (centered)

$$\left.\frac{\partial^2 f}{\partial x^2}\right|_i = \frac{f_{i+1} - 2f_i + f_{i-1}}{\Delta_x^2} + \mathcal{O}(\Delta_x^2)$$

- 2nd derivatives come up frequently both within and at edge of mesh

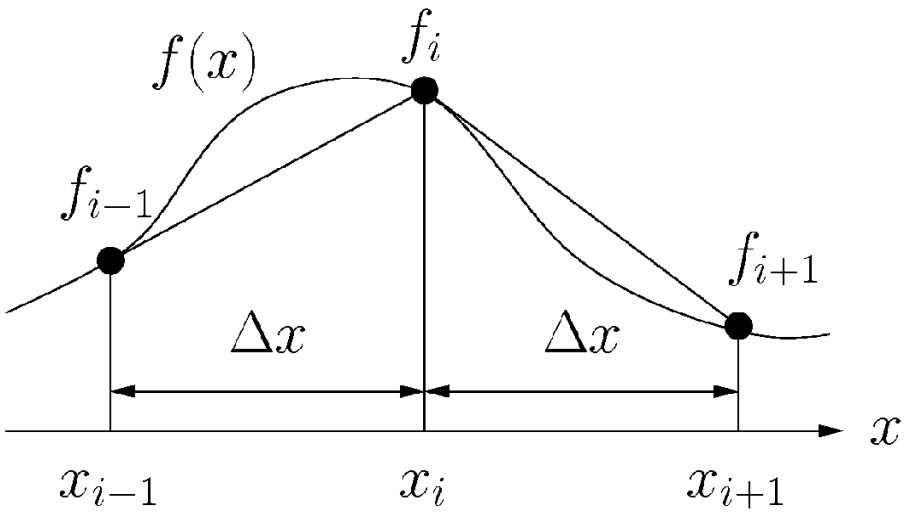
Discrete Numerical Operations: Integrals/Quadrature

Take n_x even, then $\int_{x_{min}}^{x_{max}} dx\, f(x)$ can be composed as sub-integrals of the form

$$\int_{x_{i-1}}^{x_{i+1}} dx\, f(x)$$

Trapezoidal Rule:

Using a linear approximation



$$\int_{x_{i-1}}^{x_{i+1}} dx\, f(x) = \frac{f_{i-1} + 2f_i + f_{i+1}}{2}\Delta_x + \mathcal{O}(\Delta_x^3)$$

Discrete Numerical Operations: Integrals/Quadrature (2)

Simpson's Rule:

Better approximations can be found using Taylor series expansions and the previous discrete derivatives:

$$f(x) = f_i + \left.\frac{\partial f}{\partial x}\right|_i x + \frac{1}{2!}\left.\frac{\partial^2 f}{\partial x^2}\right|_i x^2 + \cdots$$

$$= f_i + \frac{f_{i+1} - f_{i-1}}{2\Delta_x} x + \frac{f_{i+1} - 2f_i + f_{i-1}}{\Delta_x^2} x^2 + \mathcal{O}\left(\frac{|x - x_i|^3}{\Delta_x^3}\right)$$

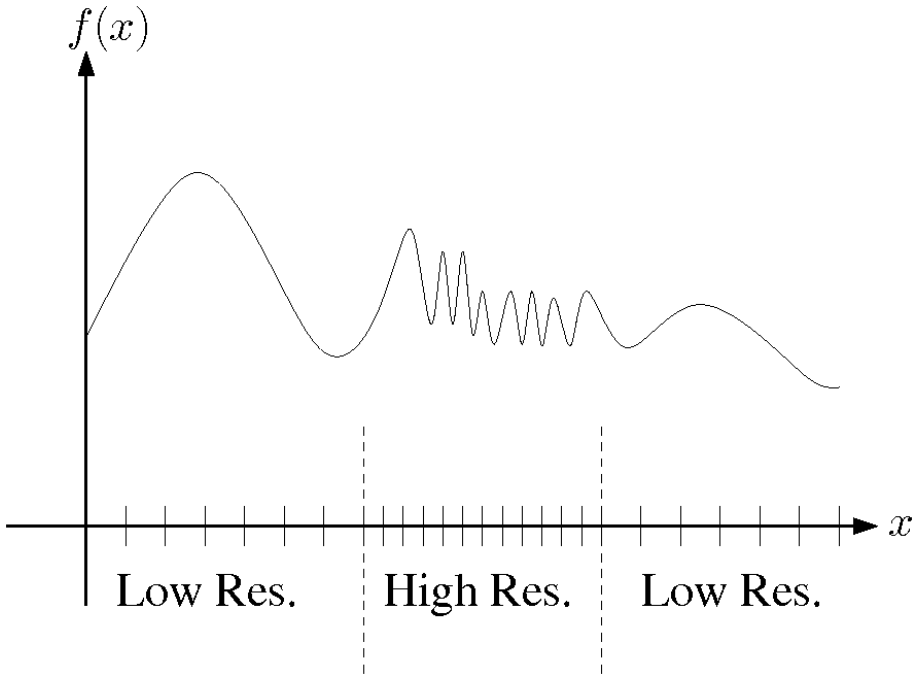
giving:

$$\int_{x_{i-1}}^{x_{i+1}} dx\, f(x) = \frac{f_{i-1} + 4f_i + f_{i+1}}{3}\Delta_x + \mathcal{O}(\Delta_x^5)$$

In the examples given, uniform grids have been employed and the formulas presented for derivatives and integrals are readily generalized to multiple dimensions.

Discrete Numerical Operations: Irregular Grids

Nonuniform grids can be used to concentrate resolution where it is needed



- Can be used most effectively when high resolution is needed only in limited regions and simulation domains are large
- Nonuniform grids make discretized formulas more complicated, particularly with respect to ordering errors
  - A simple example of nonuniform derivative calculation is included in the homework to illustrate methods

Discrete Numerical Operations: Axisymmetric Systems

In solution of axiysmmetric $(\partial/\partial\theta = 0)$ equations such as the Poisson equation on an r-z mesh

$$\nabla^2\phi(r,\theta,z) = -\frac{\rho(r,\theta,z)}{\epsilon_0}$$

$$\frac{\partial}{\partial\theta} = 0 \quad \Longrightarrow \quad \left(\frac{\partial^2}{\partial r^2} + \frac{1}{r}\frac{\partial}{\partial r} + \frac{\partial^2}{\partial z^2}\right)\phi(r,z) = -\frac{\rho(r,z)}{\epsilon_0}$$

terms like $\frac{1}{r}\frac{\partial\phi}{\partial r}$

can be difficult to accurately discretize near r=0 at the center of an axisymmetric mesh.

Ways to deal with this:

- Approx values near r=0 and live with larger errors
- Use special numerical methods
- Represent operator in x,y to avoid issue
  - Can symmetrize results to reduce errors

C: Numerical Solution of Moment Methods – Time Advance

We now have the tools to numerically solve moment methods. The moment equations may always be written as an N-dimensional set of coupled 1st order ODEs:

$$\mathbf{M} = \left(\langle x \rangle_\perp, \cdots, \langle x^2 \rangle_\perp, \cdots \right) \qquad \text{N-dim vector of moments}$$

$$\frac{d\mathbf{M}}{ds} = \mathbf{F}(\mathbf{M}, s) \qquad \text{vector equation of motion}$$

- Methods developed to advance moments can also be used for advances in particle and distribution methods

/// Example: Axisymmetric envelope equation for a continuously focused beam

$$\frac{d^2 R}{ds^2} + k_{\beta 0}^2 R - \frac{Q}{R} - \frac{\varepsilon_x^2}{R^3} = 0$$

$$R = 2\sqrt{\langle x^2 \rangle_\perp}$$

$$\frac{d}{ds}\begin{bmatrix} R \\ R' \end{bmatrix} = \begin{bmatrix} R' \\ -k_{\beta 0}^2 R + \frac{Q}{R} + \frac{\varepsilon_x^2}{R^3} \end{bmatrix}$$

$k_{\beta 0}^2,\ Q,\ \varepsilon_x^2,$ constants

///

C: Numerical Solution of Moment Methods – Euler Advance

Euler's Method:

Apply the forward difference formula

$$\left.\frac{d\mathbf{M}}{ds}\right|_i = \frac{\mathbf{M}_{i+1} - \mathbf{M}_i}{\Delta_s} + \mathcal{O}(\Delta_s) = \mathbf{F}(\mathbf{M}_i, s_i)$$

Rearrange to obtain 1st order Euler advance:

$$\mathbf{M}_{i+1} = \mathbf{M}_i + \mathbf{F}(\mathbf{M}_i, s_i)\Delta_s + \mathcal{O}(\Delta_s^2)$$

- Moments advanced in discrete steps in s from initial values

Note that N_s steps will lead to a total advance error

$$\text{advance error} \sim N_s \cdot \mathcal{O}(\Delta_s^2) \sim \frac{s_{max} - s_{min}}{\Delta_s}\mathcal{O}(\Delta_s^2) \sim \mathcal{O}(\Delta_s)$$

- Advance error decreases only linearly with step size
- Numerical work for each step is only one evaluation of **F**

C: Numerical Solution of Moment Methods – Order Advance

Definition:

A discrete advance with step error $\mathcal{O}(\Delta_s^n)$ is called an (n-1)th order method

- Euler's method is a 1st order method
- Higher order methods are generally used for ODE's in moment methods
  - Numerical work to evaluate **F** small: no need to limit evaluations
- Low order methods are generally used for particle and distribution methods
  - Numerical work to evaluate **F** large: want to minimize evaluations

C: Numerical Solution of Moment Methods – Runge-Kutta Advance

Runge-Kutta Method:

Integrate from s_i to s_{i+1} :

$$\frac{d\mathbf{M}}{ds} = \mathbf{F}(\mathbf{M}, s)$$

$$\mathbf{M}_{i+1} = \mathbf{M}_i + \int_{s_i}^{s_{i+1}} ds\, \mathbf{F}(\mathbf{M}, s)$$

Approximate $\mathbf{F}$ with a Taylor expansion through the midpoint of the step, $s_{i+1/2}$

$$\mathbf{F}(\mathbf{M}, s) = \mathbf{F}(\mathbf{M}_{i+1/2}, s_{i+1/2}) + \left.\frac{\partial \mathbf{F}}{\partial s}\right|_{s_{i+1/2}} (s - s_{i+1/2}) \quad + \quad \Theta(\Delta_s^2)$$

The linear term integrates to zero, leaving

$$\Rightarrow \mathbf{M}_{i+1} = \mathbf{M}_i + \mathbf{F}(\mathbf{M}_{i+1/2}, s_{i+1/2}) \cdot \Delta_s + \mathcal{O}(\Delta_s^3)$$

Runge-Kutta Advance (2)

Note: only need $\mathbf{M}_{i+1/2}$ to $\mathcal{O}(\Delta_s^2)$ for $\mathcal{O}(\Delta_s^3)$ accuracy, so we can apply Euler's method for the two-step procedure:

2nd Order Runge-Kutta Method:

Step 1: $$\mathbf{K} = \mathbf{F}(\mathbf{M}_i, s_i)\Delta_s$$

Step 2: $$\mathbf{M}_{i+1} = \mathbf{M}_i + \mathbf{F}\left(\mathbf{M}_i + \frac{\mathbf{K}}{2}, s_i + \frac{\Delta_s}{2}\right)\Delta_s + \mathcal{O}(\Delta_s^3)$$

- Requires *two* evaluations of $\mathbf{F}$ per advance
- 2nd order accurate in Δ_s

Higher order Runge-Kutta schemes are derived analogously from various quadrature formulas. Such formulas are found in standard numerical methods texts

- Typically, methods with step error $\mathcal{O}(\Delta_s^{N+1})$ will require N evaluations of $\mathbf{F}$

C: Numerical Solutions of Moment Methods

Many methods are employed to advance moments and particle orbits.

A general survey of these methods is beyond the scope of this lecture. But some general comments can be made:

- Many higher-order methods with adaptive step sizes exist that refine accuracy to specified tolerances and are optimized for specific classes of equations
  - Packages such as Mathematica and SciPy have many examples
- Choice of numerical method often relates to numerical work and stability considerations
- Certain methods can be formulated to exactly preserve relevant single-particle invariants
  - "Symplectic" methods preserve Hamiltonian structure of dynamics: This is very important for long advances to model particles evolving in rings over many laps.
- Accelerator problems can be demanding due to multiple frequency scales and long tracking times/distances
  - Hamiltonian dynamics; phase space volume does not decay

C: Numerical Solutions of Moment Methods – Numerical Stability

"Numerical Reversibility" test of stability:

In this method, the final value of an advance is used as an initial condition. Then the problem is run backwards to the original starting point and deviations from the initial conditions taken in the original advance are analyzed.

- Provides a simple, but stringent test of accuracy
- Will ultimately fail due to roundoff errors and cases where there is a sensitive dependence on initial conditions
  - Chaotic orbits a common example
- Orbits can be wrong but qualitatively right
  - Lack of convergence does not necessarily mean results will be useless
  - Right "pattern" in chaotic structures can be obtained with inaccurate orbits
  - Will quantify better later

We will now briefly overview an application of moment equations, namely the KV envelope equations, to a practical high current transport lattice that was designed for Heavy Ion Fusion applications at Lawrence Berkeley National Laboratory.

C: Moment Equation Application: Perp. KV Envelope Eqns

Neglect image charges and nonlinear self-fields (emittance constant) to obtain moment equations for the evolution of the beam envelope radii

$$\frac{d^2 r_x}{ds^2} + \kappa_q r_x - \frac{2Q}{r_x + r_y} - \frac{\varepsilon_x^2}{r_x^3} = 0 \qquad r_x = 2\sqrt{\langle x^2 \rangle_\perp}$$

$$\frac{d^2 r_y}{ds^2} - \kappa_q r_y - \frac{2Q}{r_x + r_y} - \frac{\varepsilon_y^2}{r_y^3} = 0 \qquad r_y = 2\sqrt{\langle y^2 \rangle_\perp}$$

$$Q = \frac{qI}{2\pi\epsilon_0 m c^3 \gamma_b^3 \beta_b^3}$$

Dimensionless Perveance
measures space-charge strength

$$\varepsilon_x = 4\left[\langle x^2 \rangle_\perp \langle x'^2 \rangle_\perp - \langle x x' \rangle_\perp^2\right]^{1/2}$$

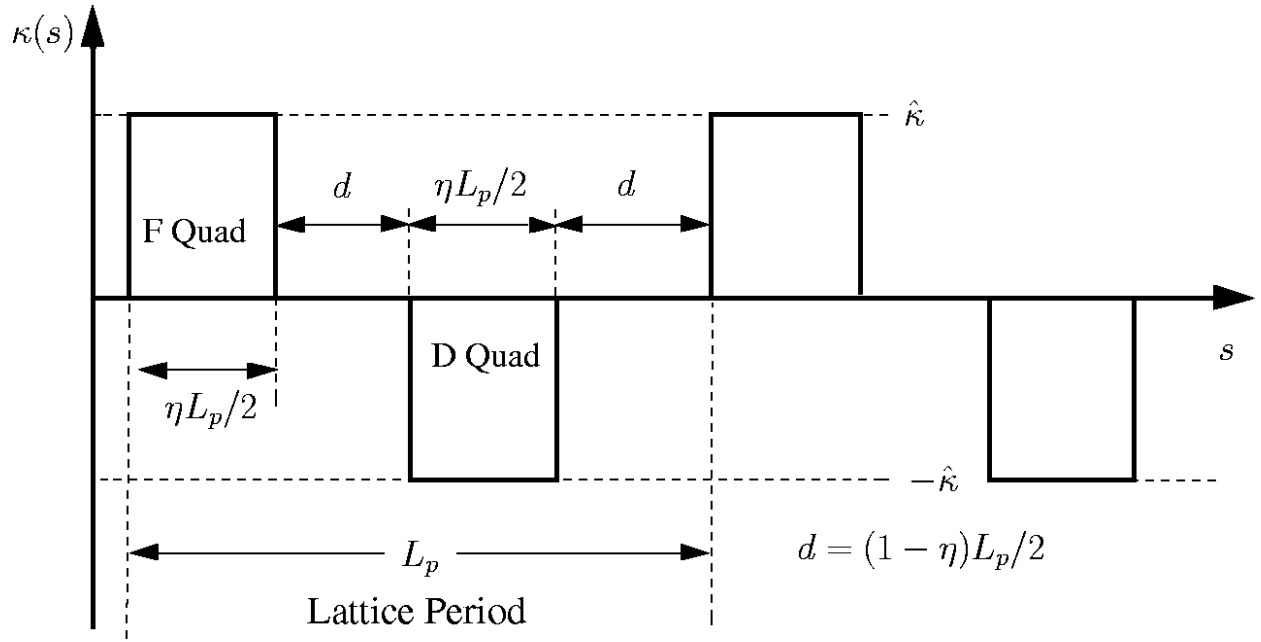
$(\varepsilon_{xn} = \gamma_b \beta_b \varepsilon_x$ normalized)

RMS Edge Emittance
measures x-x' phase-space area
~(beam size)sqrt(thermal temp.)

The matched beam solution together with parametric constraints from engineering, higher-order theory, and simulations are used to design the lattice.

Application Example Continued (2) – Focusing Lattice

Take an alternating gradient FODO doublet lattice



$$d = (1-\eta)\frac{L_p}{2} \qquad \eta = \text{Quadrupole Occupancy } (0 < \eta \leq 1)$$

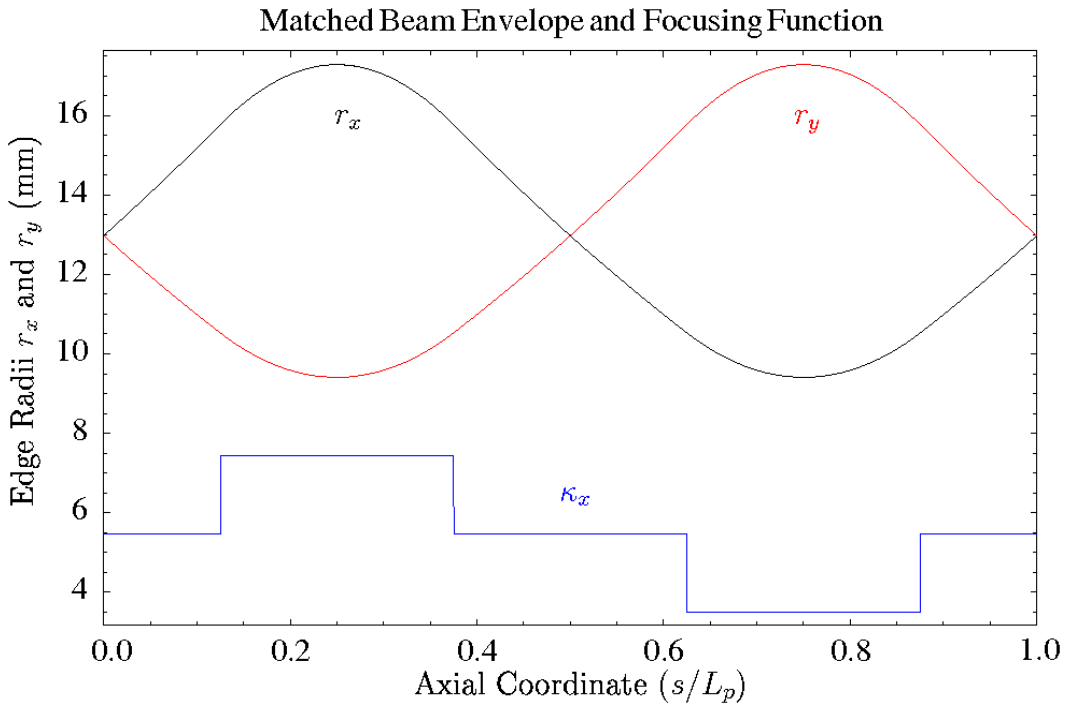
Focusing Strength

$$\hat{\kappa} = \begin{cases} \dfrac{1}{[B\rho]}\left|\dfrac{dB_x}{dy}\right|, & \text{Magnetic Quadrupole} \\ \dfrac{1}{[B\rho]\beta_b c}\left|\dfrac{dE_x}{dy}\right|, & \text{Electric Quadrupole} \end{cases}$$

Rigidity

$$[B\rho] = m\gamma_b\beta_b c/q$$

Application Example Contd. (3) – Matched Envelope Properties



Ion

K^+, E = 2 MeV

Current

I = 800 mA

Lattice

$L_p = 0.5\text{m}$

$\eta = 1/2$

$\sigma_0 = 80^\circ/\text{Lattice Period}$

Envelope Properties:

1) Low Emittance Case: $\varepsilon_x = \varepsilon_y = 50$ mm-mrad; $\sigma = 9.42^\circ/\text{Lattice Period}$

$\text{Max}[r_x] = \text{Max}[r_y] = 17.3$ mm $\quad$ $\text{Max}[r_x'] = -\text{Min}[r_y'] = 47.5$ mrad

$\text{Min}[r_x] = \text{Min}[r_y] = 9.41$ mm $\quad$ $\text{Max}[r_y'] = -\text{Min}[r_x'] = 47.5$ mrad

2) High Emittance Case: $\varepsilon_x = \varepsilon_y = 200$ mm-mrad; $\sigma = 32.13^\circ/\text{Lattice Period}$

$\text{Max}[r_x] = \text{Max}[r_y] = 18.9$ mm $\quad$ $\text{Max}[r_x'] = -\text{Min}[r_y'] = 52.4$ mrad

$\text{Min}[r_x] = \text{Min}[r_y] = 10.1$ mm $\quad$ $\text{Max}[r_y'] = -\text{Min}[r_x'] = 52.4$ mrad

Corrections and suggestions for improvements welcome!

These notes will be corrected and expanded for reference and for use in future editions of US Particle Accelerator School (USPAS) and Michigan State University (MSU) courses. Contact:

Prof. Steven M. Lund
Facility for Rare Isotope Beams
Michigan State University
640 South Shaw Lane
East Lansing, MI 48824

lund@frib.msu.edu
(517) 908 – 7291 office
(510) 459 - 4045 mobile

Please provide corrections with respect to the present archived version at:

https://people.nscl.msu.edu/~lund/uspas/scs_2016

Redistributions of class material welcome. Please do not remove author credits.

References:

Numerical Methods

Forman S. Acton, *Numerical Methods that Work*, Harper and Row Publishers, New York (1970)

Steven E. Koonin, *Computational Physics*, Addison-Wesley Publishing Company (1986)

W. Press, B. Flannery, S. Teukolsky, W. Vetterling, *Numerical Recipes in C: The Art of Scientific Computing*, Cambridge University Press (1992)

Particle Methods

C.K. Birdsall and A.B. Langdon, *Plasma Physics via Computer Simulation*, McGraw-Hill Book Company (1985)

R.W. Hockney and J.W. Eastwood, *Computer Simulation using Particles*, Institute of Physics Publishing (1988)

Review of Initial Distribution Loads

S. Lund, T. Kikuchi, and R. Davidson, "Generation of initial kinetic distributions for simulation of long-pulse charged particle beams with high space-charge intensity," PRSTAB **12**, 114801 (2009)