Overall Content Structure

Lecture Goals

- Understand DG spatial discretization (advective)
  - DG weak form
    - Test function
      - to minimize residual
      - or test function orthogonal
    - Solution approximation — and initial conditions
    - Mapping — Physical -> Computation domain — for curvilinear domains
    - DG Galerkin formulation
    - Integration by parts -> Flux functions — Solution smoothness requirements: — Differences from FEM
    - Linear vs. Non-linear flux — Ramifications — for semi-discrete system
    - Hyperbolic vs. Parabolic
    - Applying BCs — include periodic BCs
- Understand time discretization
- Method of lines style semi-discrete form
  - Types, e.g.
    - Forward Euler
    - RK4
  - CFL condition & stability
- Learn how to apply DG to arbitrary PDEs & realm of applicability
  - Intuitive understanding of methodology
  - Conceptualization of process — not tied down to specific examples
  - Understand pros / cons
  - Understand how DG "simplifies" to
    - FVM
    - FEM
- Generate runnable code of your own
  - Self-contained set of knowledge & algorithms — to be able to write a full solver

Assumed Prerequisite Knowledge

- Interested viewer: advanced -
  - Undergrad
  - Graduate
- STEM background
  - ODEs / PDEs
  - Linear Algebra
  - Calculus
- Basic programming background
  - e.g. Matlab
- Coursed taken(assumed)
  - Solution of PDEs
  - Numerical Methods
- About common underlying techniques
  - e.g. Interpolation
  - Not intended to teach
  - But may [Recall] their important features

Numerical Methods Prerequisites

- Linear algebra
  - Vector spaces, bases, properties, etc.
  - Orthogonality
  - Maybe some useful spaces, e.g.:
    - Hilbert
    - Square integral
- Polynomial interpolation (ID and 2D)
  - Lagrange, Hermite
  - Monomial basis — & ill-conditioned nature
  - Orthogonal basis
    - Legendre
    - Chebyshev
  - L2 projection
  - Choice of interpolation points, e.g.:
    - Equispaced
    - GL
    - LGL
  - Runge phenomenon
  - Vandermonde matrix — Transformation — modal -> nodal spaces
- Quadrature (ID and 2D)
  - Newton-Cotes
  - Gauss/Hermite(Legendre)
  - Relation to interpolation
- Solution of ODEs
  - Forward Euler
  - RK4
  - Implicit schemes — e.g. Backward Euler
  - Stability, Convergence

Solutions of PDEs Prerequisites

Domain representation		Meshing
	BCs	Neumann
		Dirchlet
Finite Difference Methods (FDM)		Pointwise spatial derivatives
		Computational vs. Physical domains
		Basic mapping (bilinear)
Finite Volume Methods (FVM)		Flux functions
		Artificial viscosity
		Linear vs. Non-linear fluxes
Finite Element Methods (FEM)		Weak and strong form formulation
		Piecewise linear solution approximation
		Galerkin style test functions
		Local support

Topics Layout

Module 1: What is DG?	Module 2: A Simple ID DG Solver	Module 3: To Higher-Orders (nodal) 3A: Sol'n Approximation	Module 3: To Higher-Orders (nodal) 3B: Discrete System
DG motivation (why vs. FEM, FVM, FDM)	Linear solution approximation	Revisit weak form	Numerical Quadrature (Gauss)
	Test function choice (Galerkin)	Approx. space	Hermite interpolation (2N+1 quad)
Scalar conservation law (linear) PDE	Upwind flux	L2 Projection minimizes residual norm	Truncation error/exact quadrature
	Mass Matrix	Test space -> orthogonal	GL Lagrange orthogonality
Weak form derivation	Stiffness Matrix	Monomial basis?	Local differentiation
	Putting it all together (linear system)	Ill-conditioning of monomials	Flux interpolation
Global domain vs. local element	Semi-discrete system	Recall: Lagrange interpolation (code)	Stiffness Integral
	Forward Euler	Derive Lagrange spatial approximation	Numerical Flux (interpolated)
Multiple-valued element boundaries	Investigate h-convergence	Equispaced interp points?	Assembly of system
	Investigate t-convergence	Runge phenomenon	RK4 time discretization
Recall: Flux functions	Investigate stability (CFL)	Why: Bernstein/Markov inequality	Investigate p-convergence (smoothness reqs)
		Roots of Leg instead	

A Pedagogical Comment

Take advantage of format	Each section may have subsectionsbut overall section is intended to be a self-contained concept.	Easy to "zone-out", before the start of new section	Each Module has a larger self-contained concept.
replay pause speed up slow down	First slide of a new section has the title format Section: Subsection	try and put what you learned into action: Make a code snippet — to test your understanding / or verify a claimed result etc.	Should be able to put together a script that accomplishes something substantial!