2022/09/22

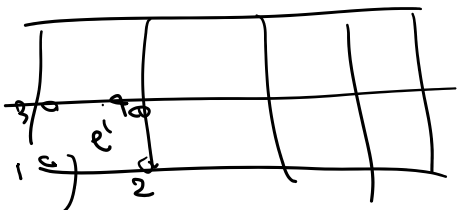
Thursday, September 22, 2022 9:44 AM

As mentioned last time, mass lumping makes the solution trivial for CFEMs and in fact the mass matrix becomes diagonal

$M = C = \sum_e \int N^T \rho N \, dV$ consistent mass / capacitance

$M^e = \begin{bmatrix} m_1 & & & 0 \\ & m_2 & & \\ & & m_3 & \\ 0 & & & m_4 \end{bmatrix}$

eg in the geometry shown



$M^e = \frac{\rho L_x L_y}{4} \begin{bmatrix} 1 & & & 0 \\ & 1 & & \\ & & 1 & \\ 0 & & & 1 \end{bmatrix}$

For the discussion above, it is clear why the majority of DG methods for hyperbolic and parabolic PDEs use explicit solvers so that can take advantage of their block diagonal LHS matrix.

CFEM:
It can be argued that for CFEMs we get the same effect by mass lumping.
This is true and in fact almost always mass lumping is done with explicit solvers for CFEMS, otherwise there is no advantage in using an explicit method (a big sparse matrix like implicit solvers but without implicit solver advantages)

Especially, we note that explicit integrators tend to SHORTEN the frequency.
Mass Lumping has the opposite effect.

This is a "match made in heaven", because not inly mass lumping results in a diagonal M for CFEM, but also do this by counteracting frequency correcting effect.
This improves dispersion errors.

The only disadvantage is losing the order of accuracy with mass lumping.

Another MAJOR advantage of an explicit method is that we don't need to assemble the stiffness matrix because it's contribution moves to the RHS.

$C\dot{A} + KA = F$

explicit solve

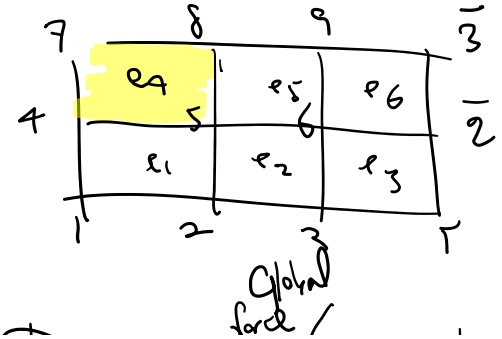
$C A_{n+1} = F_{n+1} = \Delta t (F_n - K A_n) + C A_n$ (from previous step)

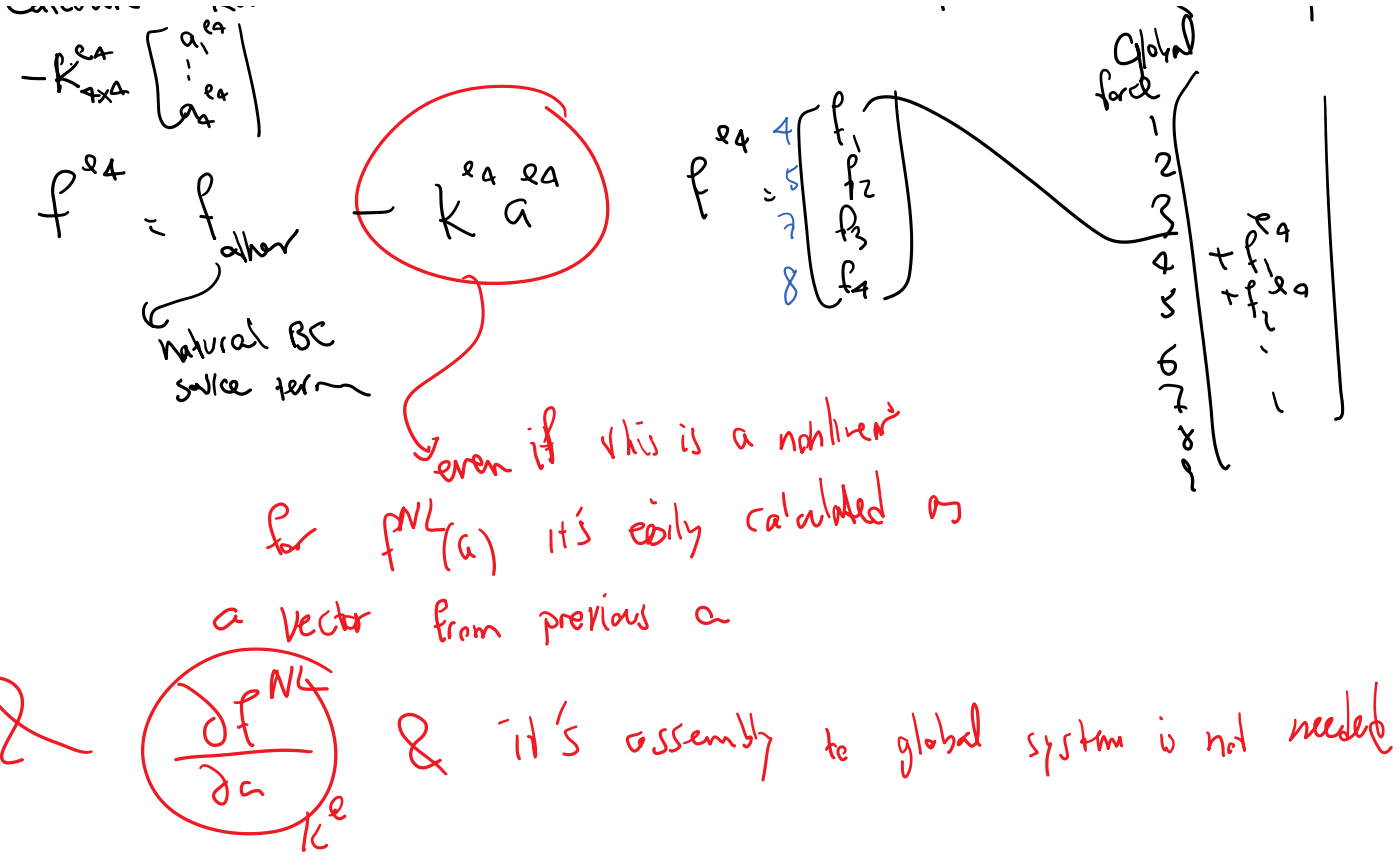
We can eliminate the need to calculate the global stiffness matrix and DIRECTLY calculate Ka_n

$Map(e_4) = [4 \; 5 \; 7 \; 8]$

Calculate Ka at element level

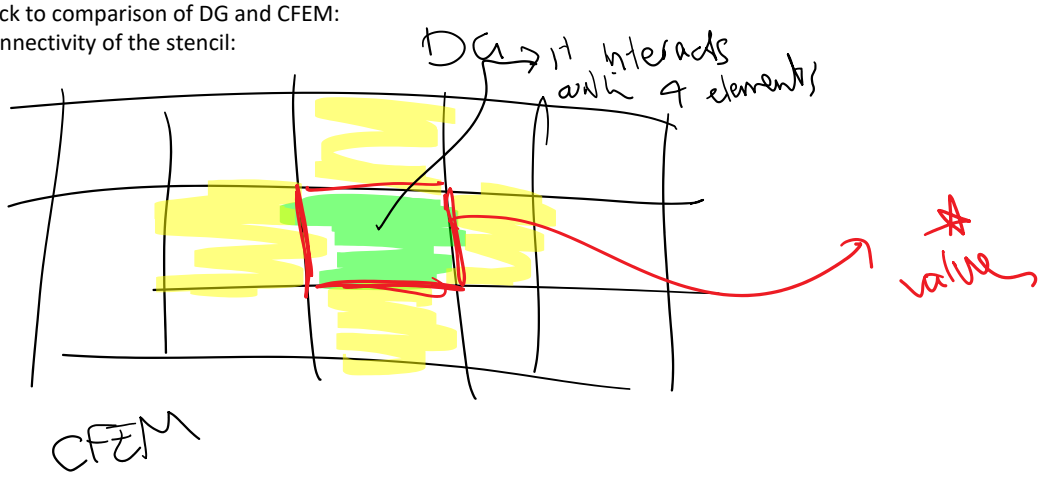
$-K^{e_4}_{4\times 4} \begin{bmatrix} a_1^{e_4} \\ \vdots \\ e_4 \end{bmatrix}$

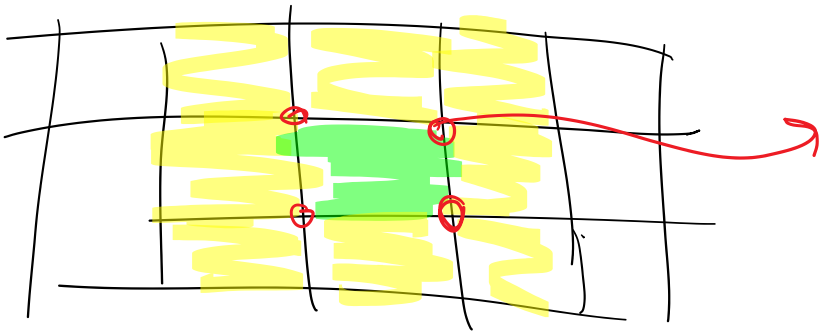




In fact, even if we had a very difficult to handle nonlinear response (nonlinear elasticity, etc.) since we don't need to compute the stiffness matrix, formulation and implementation of a time matching explicit method becomes quite simple.

Back to comparison of DG and CFEM:
Connectivity of the stencil:

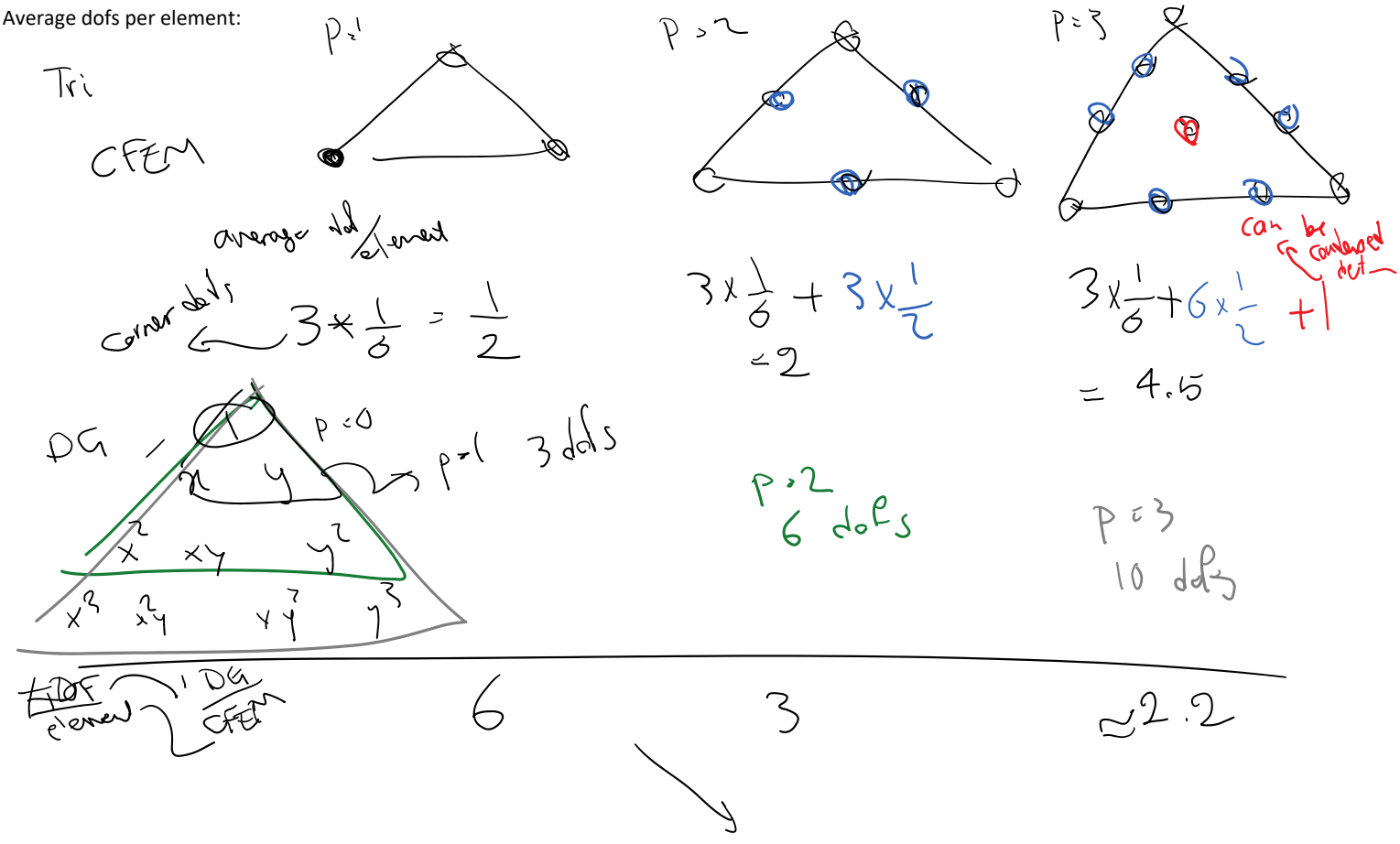




- Smaller connectivity stencil is ideal for parallel computing, because or example in domain decomposition approaches fewer pack-unpack (PUP) routines are called).
- Finite element methods are in general better than Finite Difference (FD) methods because no matter how high the order of element (-> order of accuracy) is, unlike FD methods, the stencil does not telescopically grow and remain within one neighbor element.
- DG method have an advantage in this respect because their connectivity is though the edges not the nodes -> fewer element communications.
- In this respect, tri/tet elements gain more than square and hex elements.

Note, in HW2, please skip the question on the connectivity of tet elements.

Average dofs per element:



As the polynomial order increases, the ratio of dofs of DG/CFEM decreases. So, **DG becomes better in terms of the number of unknowns as the polynomial order increases**.

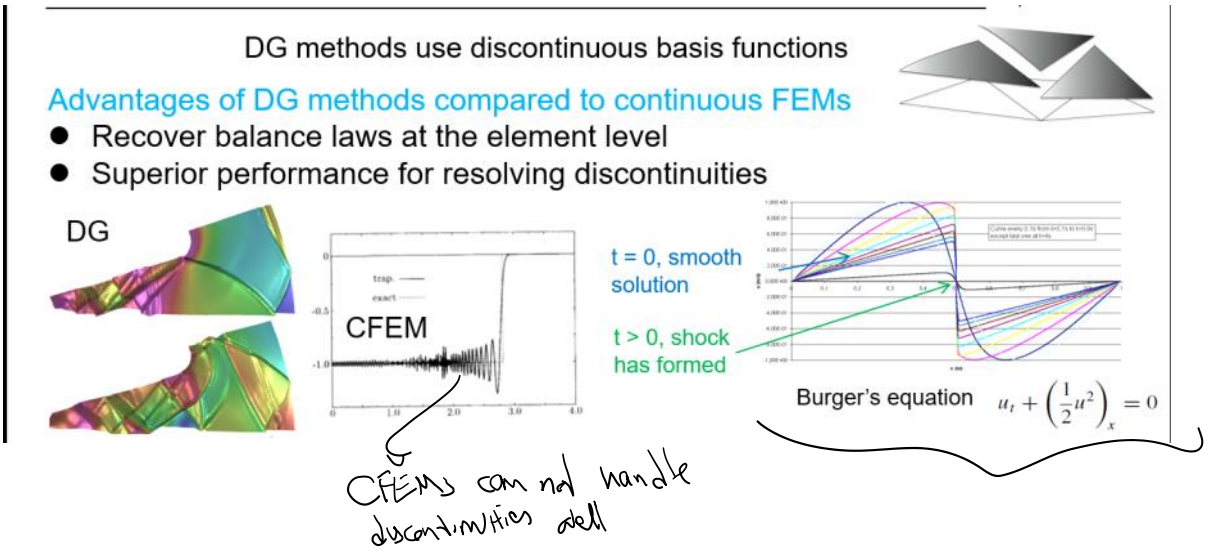
-> CFEMs in practice are often low order (often p = 1, 2), DG it's often better to use a higher order scheme.

In any case, when global system matrices are involved (e.g. an implicit solver), the interior dofs can be condensed out.

In short, DG methods have more dofs, but this ratio shrinks for higher order schemes. That's why DG is often used with high order elements.

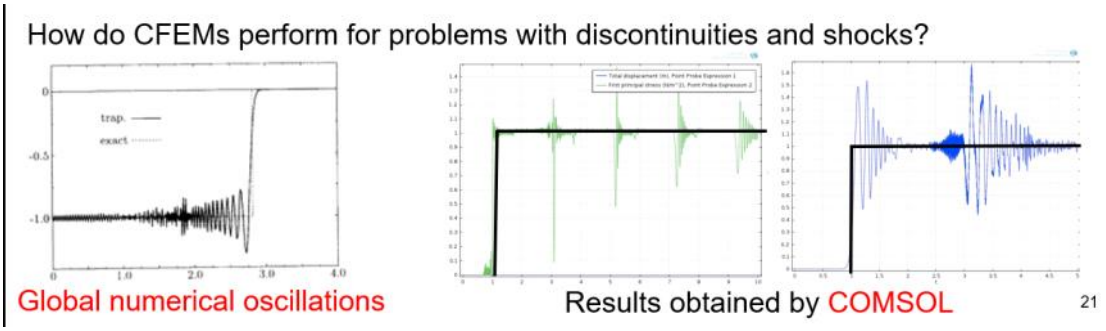
Continuation of comparing DG methods and CFEMs

Better solution accuracy and stability for problems with solution gradients and shocks for DG methods



Strong and weak shocks are common for hyperbolic PDEs:

- Linear PDEs: discontinuities are either in the IC or created by sudden application of the load on the boundary (or the source term)
- Nonlinear PDEs (e.g. Burger's equation Euler's equation for inviscid flow, etc) shocks can form even from smooth IC/BC/Source term



DG methods are very favorable for Hyperbolic PDEs

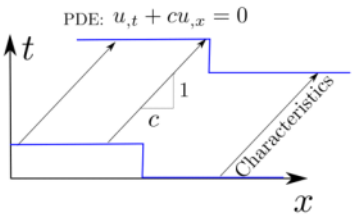
3. **Hyperbolic equations**(dynamic):

- Sample:

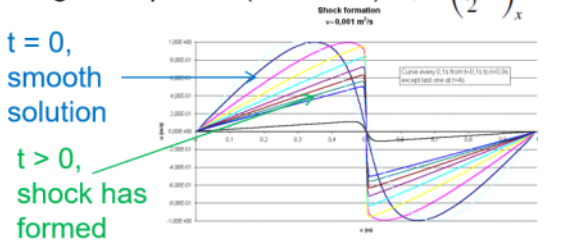
$$\rho\frac{\mathrm{d}^2\mathbf{u}}{\mathrm{d}t^2} - \nabla.\sigma = \rho\mathbf{b} \qquad \text{Elastostatics equation}$$

$$\frac{\mathrm{d}^2u}{\mathrm{d}t^2} - k\Delta u = f$$

- There is a maximum speed for the propagation of waves (information).
- Due to finiteness of the wave speed the spatial domain is NOT globally coupled.
- Numerical methods may employ the locality of hyperbolic systems to devise local solution schemes.
- Unlike parabolic equations, hyperbolic equations preserve discontinuities or even generate them (nonlinear equations).

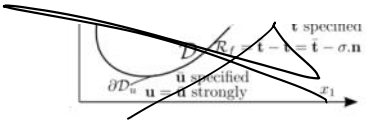


Burger's equation (nonlinear) $u_t + \left(\frac{1}{2}u^2\right)_x = 0$



12

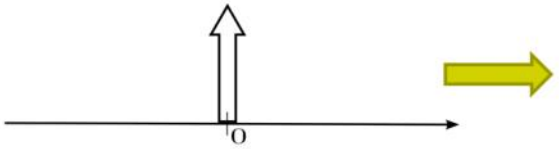
2. **Parabolic equations**(dynamic):



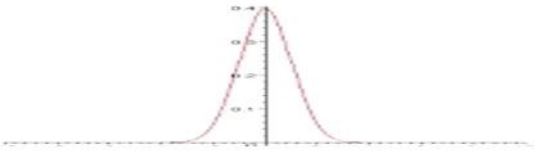
- Sample:

$$C\frac{\mathrm{d}T}{\mathrm{d}t} - \kappa\Delta T = Q, (\text{constant } \kappa), \qquad \text{Parabolic(Fickian) heat equation}$$

- Imply an infinite speed of propagation of information.
- The entire spatial domain is coupled.
- Numerical methods may involve the solution of the global spatial domain or local domains.
- The diffusive operator **smoothens the solution**.



$$T(x, t = 0) = \delta(x)$$



$$G(x,t) = \sqrt{\frac{C}{4\pi\kappa t}}\exp\left(-\frac{Cx^2}{4\kappa t}\right) \quad \text{Green's function}$$

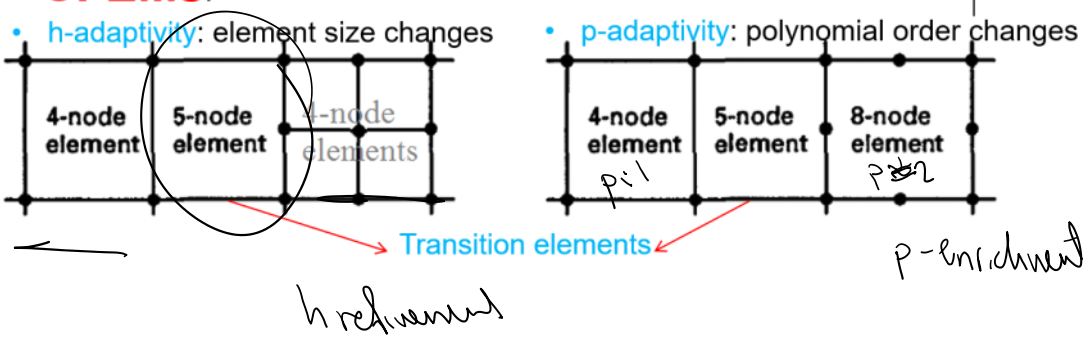
11

For parabolic PDEs, the better handling of shocks is not a concern and DG methods are not more advantageous from this perspective.

Adaptivity

CFEMs:

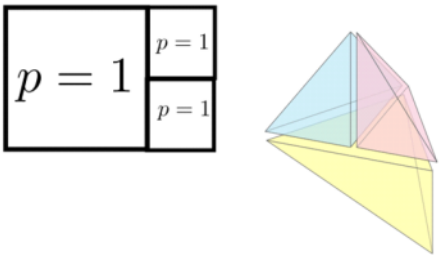
- h-adaptivity: element size changes
- p-adaptivity: polynomial order changes

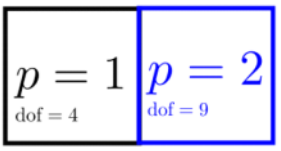


This becomes intractable for higher order elements and/or more refinement levels.

DGs:

- h-adaptivity:
- p-adaptivity:





Arbitrary change in size and polynomial order as jump conditions are weakly enforced

17

Summary of CFEMs and DG methods

Advantages of DG methods:

1. FEM adaptivity
   Resolving shocks and discontinuities for hyperbolic problems
   Recovering balance laws at the element level
   } no transition elements needed

2. Efficiency /dynamic problems (block diagonal "mass" matrix)
3. Parallel computing (more local communication and use of higher order elements with DG methods)
4. Superior performance for resolving discontinuities (discrete solution space better resembles the continuum solution space)

Arbitrary change in size and polynomial order

5. Can recover balance properties at the element level (vs global domain)

Disadvantages:

- Higher number of degrees of freedom:
  - Particularly important for elliptic problems (global system is solved).
  - Recently hybridizable DG methods (HDG), use Schur decomposition (static condensation) to eliminate elements internal dofs, making DG methods competitive or even better for elliptic problems as well.

24