Matrix-Free Preconditioners for High-Order Finite Elements on Triangles via the Duffy de Rham Complex

Will Pazner

pazner@pdx.edu

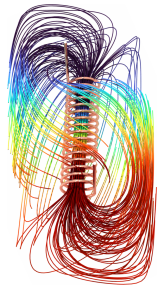
FEM@LLNL Seminar
Lawrence Livermore National Laboratory
November 9, 2026

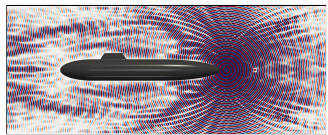
Portland State University
Fariborz Maseeh Department of
Mathematics and Statistics

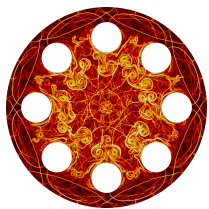
High-Order Finite Elements

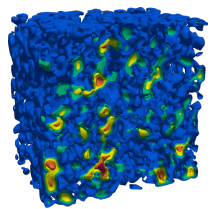
- **ACCURACY** **High-order** methods promise **higher accuracy** per degree of freedom
- **EFFICIENCY** They are **well-suited** for **modern supercomputing architectures**
- **APPLICATIONS** Inertial confinement fusion, shock hydrodynamics, seismic modeling, wave propagation
- **CHALLENGE** Realizing these benefits requires **efficient matrix-free solvers**







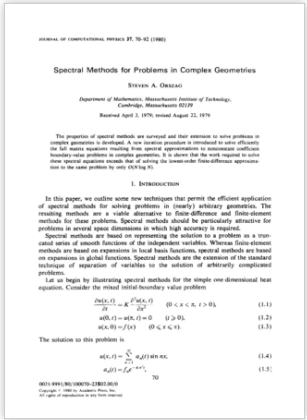




Comparison: h and p refinement

	h Mesh size $h \to 0$	p Degree $p \to \infty$	
Error Rate	Algebraic (h^k for fixed k)	Exponential (a^{-p})	😀
Arithmetic Intensity	Fixed (often low)	Tunable (often high)	😀
Geometric Representation	Less accurate	More accurate	😀
Nonzeros per Row	$\mathcal{O}(1)$	$\mathcal{O}(p^d)$	😡
(Naive) Complexity	$\mathcal{O}(n_{\text{el}})$	$\mathcal{O}(p^{3d} n_{\text{el}})$	😡
Condition Number	$\mathcal{O}(h^{-2})$	$\mathcal{O}(p^4)$	😡

Tackling the challenges: Orszag (1980)



JOURNAL OF COMPUTATIONAL PHYSICS 37, 70–92 (1980)

Spectral Methods for Problems in Complex Geometries

STEVEN A. ORSZAG

Department of Mathematics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139

Received April 3, 1979; revised August 22, 1979

The properties of spectral methods are surveyed and their extension to solve problems in complex geometries is developed. A new iteration procedure is introduced to solve efficiently the full matrix equations resulting from spectral approximations to nonconstant coefficient boundary-value problems in complex geometries. It is shown that the work required to solve these spectral equations exceeds that of solving the lowest-order finite-difference approximation to the same problem by only $O(\ln \log N)$.

1. INTRODUCTION

In this paper, we outline some new techniques that permit the efficient application of spectral methods for solving problems in (nearly) arbitrary geometries. The resulting methods are a viable alternative to finite-difference and finite-element methods for these problems. Spectral methods should be particularly attractive for problems in several space dimensions in which high accuracy is required.

Spectral methods are based on representing the solution to a problem as a truncated series of smooth functions of the independent variables. Whereas finite-element methods are based on expansions in local basis functions, spectral methods are based on expansions in global functions. Spectral methods are the extension of the standard technique of separation of variables to the solution of arbitrarily complicated problems.

Let us begin by illustrating spectral methods for the simple one-dimensional heat equation. Consider the mixed initial-boundary value problem

$$\frac{\partial u(x,t)}{\partial t} = K \frac{\partial^2 u(x,t)}{\partial x^2} \qquad (0 < x < \pi,\ t > 0), \tag{1.1}$$

$$u(0,t) = u(\pi,t) = 0 \qquad (t \geqslant 0), \tag{1.2}$$

$$u(x,0) = f(x) \qquad (0 \leqslant x \leqslant \pi). \tag{1.3}$$

The solution to this problem is

$$u(x,t) = \sum_{n=1}^{\infty} a_n(t) \sin nx, \tag{1.4}$$

$$a_n(t) = f_n e^{-Kn^2 t}, \tag{1.5}$$

0021-9991/80/100070-23$02.00/0

Copyright © 1980 by Academic Press, Inc.
All rights of reproduction in any form reserved.



Steven Orszag (1943–2011)

Orszag (1980). *Spectral Methods for Problems in Complex Geometries.* J. Comp. Phys.

Lineage: Orszag → Patera → Fischer → CEED

Tackling the challenges: Orszag (1980)

- Introduced **two foundational concepts**
  - Assumes tensor-product structure: quadrilateral or hexahedral elements

Tackling the challenges: Orszag (1980)

- Introduced **two foundational concepts**
  - Assumes tensor-product structure: quadrilateral or hexahedral elements

Idea 1. Sum Factorization

- Rather than form the matrix, **compute its action** using Kronecker structure
  - Operations $p^{3d} \to p^{d+1}$
  - Storage $p^{2d} \to p^{d}$
  - The method is necessarily **matrix-free**
  - We don't have access to the matrix entries (for e.g. preconditioning)

Tackling the challenges: Orszag (1980)

- Introduced **two foundational concepts**
  - Assumes tensor-product structure: quadrilateral or hexahedral elements

Idea 1. Sum Factorization

- Rather than form the matrix, **compute its action** using Kronecker structure
  - Operations $p^{3d} \to p^{d+1}$
  - Storage $p^{2d} \to p^{d}$
  - The method is necessarily **matrix-free**
  - We don't have access to the matrix entries (for e.g. preconditioning)

Idea 2. Low-Order Preconditioning

- The high-order method is **spectrally equivalent**, **independent of order**, to a related low-order discretization
  - Good preconditioners for the low-order method are also good preconditioners for the high-order method

Low-order-refined preconditioning: history

- **1980**: Second-order finite differences used as preconditioners (Orszag)
- **1985–**: Low-order finite elements used to precondition spectral methods (Canuto and Quarteroni, (1985, 1994, 2010, …); Deville and Mund (1985, 1990, …))
- **1996–1997**: Low-order finite elements to precondition high-order spectral elements (FEM–SEM; Pavarino, Windlund (1996), Casarin (1997), Fischer (1997))
- **2020–:** A major focus of my research in recent years
  - Semi-structured multigrid, ordered ILU smoothing (2020)
  - Nonconforming AMR, *hp*-refinement, DG methods (with Tz. Kolev, 2021)
  - Finite element de Rham complex (with Tz. Kolev, C. Dohrmann, 2022)
  - End-to-end GPU acceleration (with Tz. Kolev, J. Camier, 2023)
  - Saddle-point problems, grad-div, Darcy (with Tz. Kolev, P. Vassilevski, 2024)
  - Pressure robust IPDG Stokes (2025)

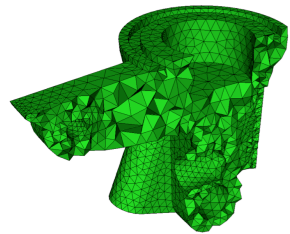
Low-order-refined preconditioning: history

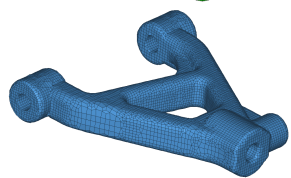
- **1980**: Second-order finite differences used as preconditioners (Orszag)
- **1985–**: Low-order finite elements used to precondition spectral methods (Canuto and Quarteroni, (1985, 1994, 2010, …); Deville and Mund (1985, 1990, …))
- **1996–1997**: Low-order finite elements to precondition high-order spectral elements (FEM–SEM; Pavarino, Windlund (1996), Casarin (1997), Fischer (1997))
- **2020–:** A major focus of my research in recent years
  - Semi-structured multigrid, ordered ILU smoothing (2020)
  - Nonconforming AMR, *hp*-refinement, DG methods (with Tz. Kolev, 2021)
  - Finite element de Rham complex (with Tz. Kolev, C. Dohrmann, 2022)
  - End-to-end GPU acceleration (with Tz. Kolev, J. Camier, 2023)
  - Saddle-point problems, grad-div, Darcy (with Tz. Kolev, P. Vassilevski, 2024)
  - Pressure robust IPDG Stokes (2025)

- **All of the above** are on tensor-product elements. **This talk:** extension to triangular elements.
  - Prior work: Olson (2007), Chalmers, Warburton (2018)

What about triangles and tetrahedra?

- Simplex meshing is much more developed than quad or hex meshing
- Automated hexahedral meshing for complex geometries is not reliable and time consuming



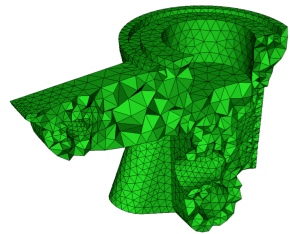


Trade-Offs for High-Order

Simplices	**Tensor-Product**
✓ Robust Automatic Meshing	✓ Sum Factorization
✓ Geometric Flexibility	✓ Low-Order Preconditioning

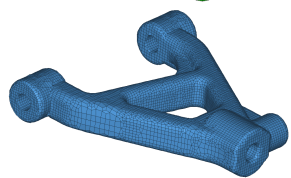
What about triangles and tetrahedra?

- Simplex meshing is much more developed than quad or hex meshing
- Automated hexahedral meshing for complex geometries is not reliable and time consuming



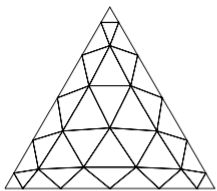
Trade-Offs for High-Order

Simplices	**Tensor-Product**
✓ Robust Automatic Meshing	✓ Sum Factorization
✓ Geometric Flexibility	✓ Low-Order Preconditioning

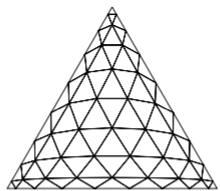


$\mathcal{Q}$: Can we recover the benefits of high-order elements on simplex meshes?

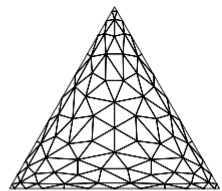
Canuto, Quarteroni, Hussaini, Zang. Spectral Methods: Evolution to Complex Geometries and Applications to Fluid Dynamics (2007).



$N = 6$



$N = 9$



$N = 12$

N	$M = 3$	$M = 6$	$M = 9$	$N = 12$
3	2.61	2.61	2.59	2.58
6	5.88	5.92	5.88	5.88

N	$M = 3$	$M = 6$	$M = 9$	$N = 12$
9	19.07	19.06	18.99	18.79
12	24.77	24.80	24.79	24.79

Fig. 5.14. *Top*: example of finite-element meshes built on the Fekete nodes in each triangular element for $N = 6, 9, 12$. *Bottom*: spectral condition numbers $\mathcal{K}(P_{3,S})$. There is no FEM-SEM spectral equivalence [Courtesy of T. Warburton, L.F. Pavarino and J.S. Hesthaven]

Background: tensor-product low-order preconditioning

- Why and how does low-order preconditioning work?
- On quads and hexes, it boils down to **1D equivalence**
- The "p version" finite element method on an interval is *spectrally equivalent* to the "h version" **with a specific mesh**

p version

Find $u_p \in \mathcal{P}_0^p$ such that for all $v_p \in \mathcal{P}_0^p$

$$\int_0^1 u_p' v_p' \, dx = \int_0^1 f v_p \, dx.$$

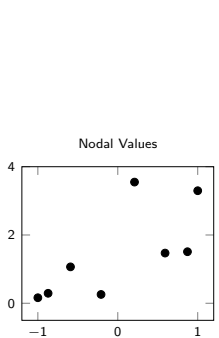
h version

$$V_h = \{v_h : [0,1] \to \mathbb{R} : v_h|_{[x_i, x_{i+1}]} \in \mathcal{P}^1\}.$$

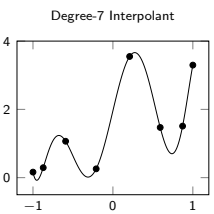
Find $u_h \in V_{h,0}$ such that for all $v_h \in V_{h,0}$

$$\int_0^1 u_h' v_h' \, dx = \int_0^1 f v_h \, dx.$$

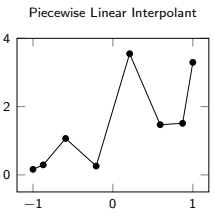
1D norm equivalence: interpolation



$\mathcal{I}_p(u)$



$\mathcal{I}_h(u)$



- $p + 1$ data points (x_i, u_i)
- x_i are Gauss–Lobatto
- L^2 norm equivalence

$$\|\mathcal{I}_p(u)\|_0 \approx \|\mathcal{I}_h(u)\|_0$$

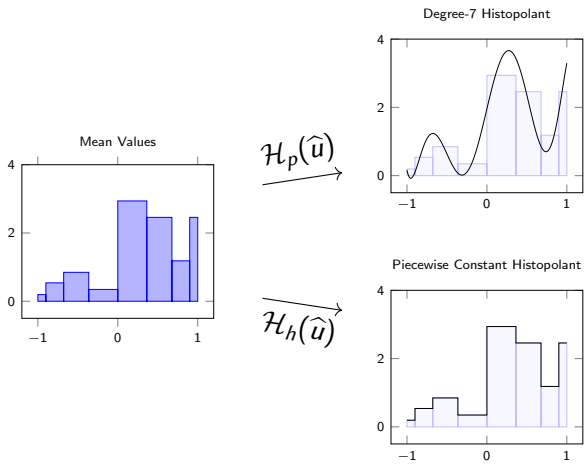
(properties of the quadrature)

- H^1 seminorm equivalence

$$\|\mathcal{I}_p(u)'\|_0 \approx \|\mathcal{I}_h(u)'\|_0$$

(H^1 **stability** of Lobatto polynomial interpolation)

1D norm equivalence: histopolation



- Subcells defined by Gauss–Lobatto points x_i

$$\int_{x_i}^{x_{i+1}} \mathcal{H}_p(\widehat{u})\, dx = \widehat{u}_i$$

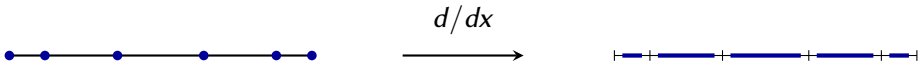
$$\int_{x_i}^{x_{i+1}} \mathcal{H}_h(\widehat{u})\, dx = \widehat{u}_i$$

- The interpolation equivalences actually imply

$$\|\mathcal{H}_p(\widehat{u})\|_0 \approx \|\mathcal{H}_h(\widehat{u})\|_0$$

as we will see

Interpolation–histopolation basis



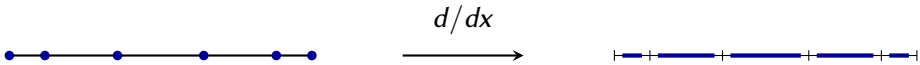
$\mathcal{Q}_p$ and V_h

Degrees of freedom are
point values (interpolation)

$\mathcal{Q}_{p-1}$ and $\widehat{V}_h$

Degrees of freedom are
subcell integrals (histopolation)

Interpolation–histopolation basis



$\mathcal{Q}_p$ and V_h

Degrees of freedom are point values (interpolation)

$\mathcal{Q}_{p-1}$ and $\widehat{V}_h$

Degrees of freedom are subcell integrals (histopolation)

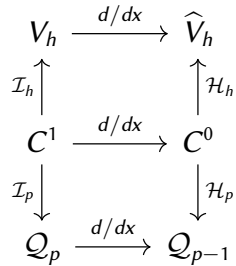
$$D_p : \mathcal{Q}_p \to \mathcal{Q}_{p-1}$$
$$D_h : V_h \to \widehat{V}_h$$

$$D_p = D_h = \begin{pmatrix} -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 \end{pmatrix}$$

- In the induced bases, the **high-order** and **low-order** differentiation matrix are identical.
- Both are given by the **signed incidence matrix**.

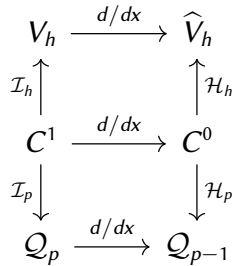
Interpolation, histopolation, and differentiation

As a consequence, the following diagram commutes.



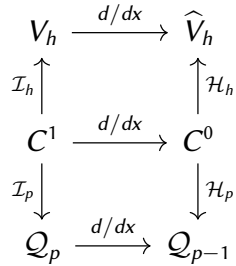
Interpolation, histopolation, and differentiation

As a consequence, the following diagram commutes.



- Given $\widehat{u}$, let w be an antiderivative, $w' = \widehat{u}$
  - $w_p := \mathcal{I}_p(w)$ is an antiderivative for $\widehat{u}_p := \mathcal{H}_p(\widehat{u})$
  - $w_h := \mathcal{I}_h(w)$ is an antiderivative for $\widehat{u}_h := \mathcal{H}_h(\widehat{u})$

As a consequence, the following diagram commutes.



- Given $\widehat{u}$, let w be an antiderivative, $w' = \widehat{u}$
  - $w_p := \mathcal{I}_p(w)$ is an antiderivative for $\widehat{u}_p := \mathcal{H}_p(\widehat{u})$
  - $w_h := \mathcal{I}_h(w)$ is an antiderivative for $\widehat{u}_h := \mathcal{H}_h(\widehat{u})$
- $\|\widehat{u}_p\|_0 = \|w_p'\|_0 \approx \|w_h'\|_0 = \|\widehat{u}_h\|_0$

Spectral equivalence

- Norm equivalences $\implies$ 1D spectral equivalence $M_p \sim M_h$ and $L_p \sim L_h$
- 2D and 3D mass and stiffness

$$M_{2\mathrm{D}} = M \otimes M, \qquad L_{2\mathrm{D}} = L \otimes M + M \otimes L$$

$$M_{3\mathrm{D}} = M \otimes M \otimes M, \qquad L_{3\mathrm{D}} = L \otimes M \otimes M + M \otimes L \otimes M + M \otimes M \otimes L$$

- Spectral equivalence follows from Kronecker properties
- Extends to general unstructured meshes through assembly

$$\widehat{A}_p \sim \widehat{A}_h \quad \implies \quad A_p = \Lambda^T \widehat{A}_p \Lambda \sim \Lambda^T \widehat{A}_h \Lambda = A_h$$

Extending this approach to triangles

- Would like to use Gauss–Lobatto nodes and proceed accordingly…

Extending this approach to triangles

- Would like to use Gauss–Lobatto nodes and proceed accordingly…

There's a Catch

The analogue to Gauss–Lobatto quadrature on triangles **is proven not to exist**



SIAM J. NUMER. ANAL.
Vol. 47, No. 2, pp. 1304–1318

© 2009 Society for Industrial and Applied Mathematics

ON THE EXISTENCE OF EXPLICIT *hp*-FINITE ELEMENT METHODS USING GAUSS–LOBATTO INTEGRATION ON THE TRIANGLE*

B. T. HELENBROOK†

Abstract. Spectral-element simulations on quadrilaterals and hexahedra rely on the Gauss-Lobatto (GL) integration rule to enable explicit simulations with optimal spatial convergence rates. In this work, it is proved that a similar integration rule does not exist on triangles. The following properties of the rule are sought: a $(p+1)(p+2)/2$ point integration rule capable of exactly integrating polynomials of up to $2p-1$ on the triangle, $p-1$ integration points located on each edge, $p-1$ integration points located on each edge, and $(p-1)(p-2)/2$ integration points located in the interior of the element. The proof hinges on the fact that a nonzero basis function exists that can be used to build a nodal basis with the appropriate number of vertex modes that can be leveled to obtain a nodal basis with $(p-1)(p-2)/2$ interior modes. The proof shows that vertex functions of a basis having this property exist and are unique, but on a triangle these functions are not nodal, and therefore the GL rule does not exist. In spite of this, the existence of the vertex functions indicates that there may be a nonnodal basis that has the above properties. The methodology explored in the paper gives insight into a possible way to find such a basis.

Key words. triangles, quadrature, integration, Gauss, Lobatto, mass-lumping

AMS subject classifications. 65N12, 74S05

DOI. 10.1137/070683439

1. Introduction. Gauss–Lobatto (GL) integration [1, p. 888] provides the foundation for spectral element simulations [20]. Not only does it provide a numerical integration method, but the integration points also define a nodal basis that allows easy enforcement of continuity constraints at element boundaries and gives an approximately diagonal mass matrix. This last point enables unsteady simulations that do not require inversion of a globally coupled mass matrix and yet still obtain optimal spatial convergence rates [20]. These properties are the main reason that spectral element simulations can efficiently achieve a high order of accuracy.

While GL rules are available for segments [1], quadrilaterals [2], quadrilaterals, and hexahedra [16, p. 143], an equivalent integration rule has not been found for triangles. This is not due to a lack of effort in searching. Much effort has been made to find optimal interpolation points on the triangle [18, 2, 3, 17, 15] and also to find a quadrature formula [28, 29, 14, 4, 5, 25, 17]. Cools and coworkers provide an excellent summary of the current state-of-quadrature rules on triangles as well as other geometries [9, 8, 10, 7, 6, 19, 11]. Because no completely satisfactory integration rule has been found, researchers are still experimenting with different techniques for performing high-order continuous finite element simulations on triangles [23, 24, 12, 30, 21].

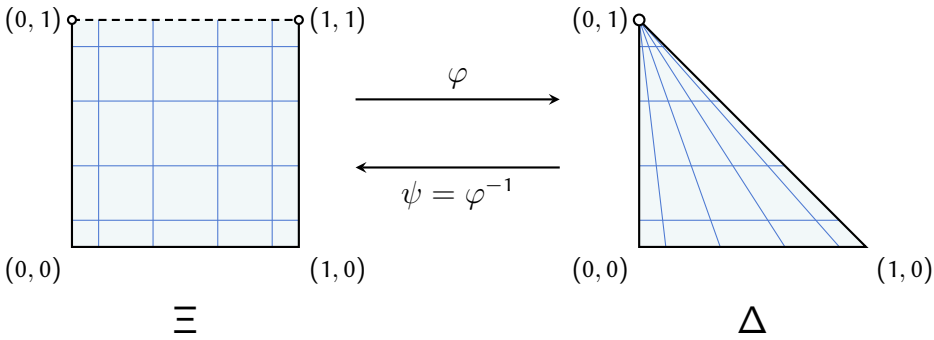
In this work, it is proved that there is no GL integration rule for a triangle that has properties similar to those for segments, quadrilaterals, and hexahedra.

*Received by the editors March 10, 2007; accepted for publication (in revised form) October 31, 2008; published electronically February 25, 2009. This material is based upon work supported by the National Science Foundation under grant 0511580.
http://www.siam.org/journals/sinum/47-2/68343.html
†Mechanical & Aeronautical Engineering Department, Clarkson University, 8 Clarkson Avenue, Potsdam, NY 13699-5725 (helenbrb@clarkson.edu).

1304

Helenbrook (2009). On the existence of explicit *hp*-finite element methods using Gauss–Lobatto integration on the triangle.

Duffy Transformation



$$\varphi(x,y) = \big(x(1-y), y\big) \qquad \psi(x,y) = \left(\frac{x}{1-y}, y\right)$$

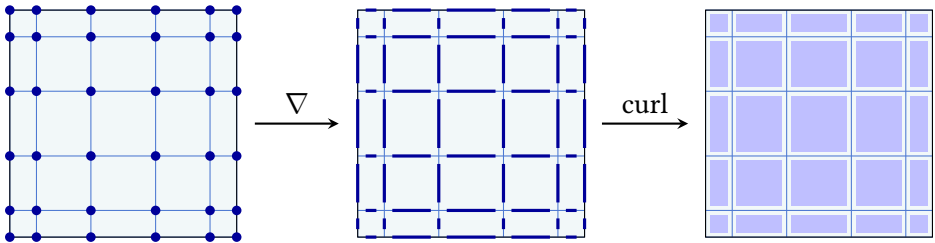
Duffy transformation

- First used for numerical quadrature by M. Duffy in 1982
- Stroud conical quadrature (1971) is Gauss–Jacobi quadrature on the square mapped under the Duffy transformation
- Used by Karniadakis and Sherwin (1995) to define orthogonal bases on simplices
- Similar approach used by Nigam and Phillips (2012) to define high-order elements on pyramids

Our goal: use favorable properties of the Gauss–Lobatto interpolation points on the square to obtain norm equivalence properties on Duffy-mapped Gauss–Lobatto points on the triangle

Interoplation–histopolation on the square

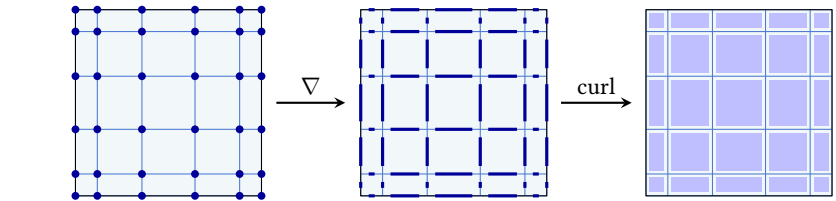
DOFs for the high-order FEM de Rham complex can be defined on the square lattice



H^1 | $H(\mathrm{curl})$ (or $H(\mathrm{div})$) | L^2

Vertex values | Edge integrals | Cell integrals

Discrete differential operators



Discrete gradient

$G_p : \mathcal{Q}_p \to \mathcal{Q}_{p-1,p} \times \mathcal{Q}_{p,p-1}$

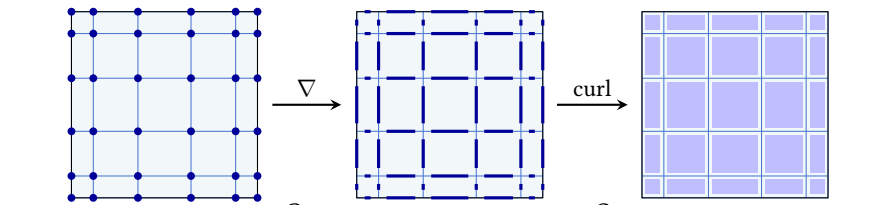
$G_h : V_h \to W_h$

Discrete curl

$C_p : \mathcal{Q}_{p-1,p} \times \mathcal{Q}_{p,p-1} \to \mathcal{Q}_{p-1}$

$C_h : W_h \to Z_h$

Discrete differential operators



Discrete gradient

$G_p : \mathcal{Q}_p \to \mathcal{Q}_{p-1,p} \times \mathcal{Q}_{p,p-1}$

$G_h : V_h \to W_h$

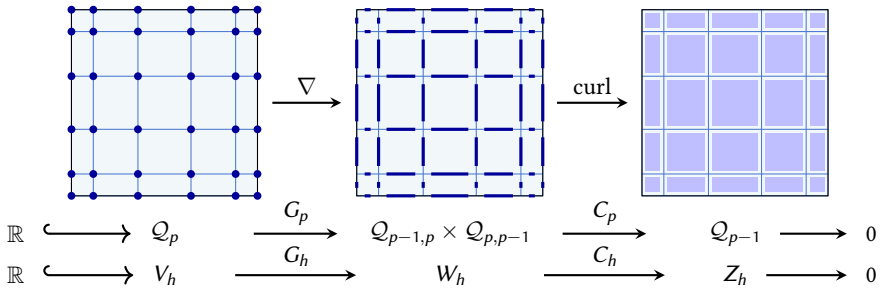
Discrete curl

$C_p : \mathcal{Q}_{p-1,p} \times \mathcal{Q}_{p,p-1} \to \mathcal{Q}_{p-1}$

$C_h : W_h \to Z_h$

$G_p = G_h$ and $C_p = C_h$, **signed incidence matrices of the lattice connectivity graph.**

Discrete differential operators



The sequences are **exact** meaning that the image of each map is exactly the kernel of the next one.

- $\mathrm{im}(G_*) = \ker(C_*)$
- C_* is surjective

L^2 equivalence $\implies$ spectral equivalence

In this setting, spectral equivalence of the stiffness matrix follows from two properties:

(P1) Discrete equality of the high-order and low-order differentiation matrix

(P2) L^2 equivalence on the target space

L^2 equivalence $\implies$ spectral equivalence

In this setting, spectral equivalence of the stiffness matrix follows from two properties:

(P1) Discrete equality of the high-order and low-order differentiation matrix

(P2) L^2 equivalence on the target space

Since $D_p = D_h$ as matrices and

$$A_p = D^T M_p D, \qquad A_h = D^T M_h D,$$

if $M_p \sim M_h$ then $A_p \sim A_h$.

L^2 equivalence $\implies$ spectral equivalence

In this setting, spectral equivalence of the stiffness matrix follows from two properties:

(P1) Discrete equality of the high-order and low-order differentiation matrix

(P2) L^2 equivalence on the target space

Since $D_p = D_h$ as matrices and

$$A_p = D^T M_p D, \qquad A_h = D^T M_h D,$$

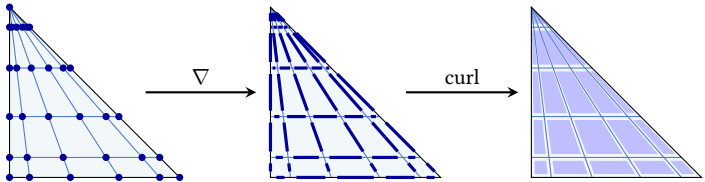
if $M_p \sim M_h$ then $A_p \sim A_h$.

(P1) holds for **any** choice of lattice points

(P2) is a **special property** of Gauss–Lobatto

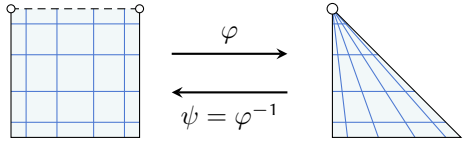
The collapsed lattice

Map the square lattice to a collapsed triangular lattice under the Duffy transformation.



- **Same procedure:** use vertex values, edge integrals, and cell integrals to define DOFs.
- **Low-order spaces:** use the standard lowest-order Lagrange, Nédélec, and L^2 elements on the mixed (quadrilateral and triangular mesh) defined by the lattice.
- **High-order space:** pull back polynomials from the square, **but:** the Duffy transformation is singular
- **Idea:** once the spaces are defined, **(P1)** will hold automatically. We will lose the strong property **(P2)**, but hope to still retain spectral equivalence.

Pullbacks

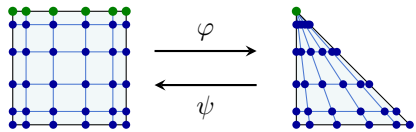


Functions defined on Ξ are mapped to functions on Δ by pulling back under $\psi = \varphi^{-1}$. Consider function in H^1, $H(\mathrm{curl})$, and L^2 as 0-, 1-, and 2-forms.

The pullbacks ("grad", "curl", and "broken") are given by:

$$\psi_g^* v = v \circ \psi \in H^1,$$
$$\psi_c^* \boldsymbol{w} = (D\psi)^T \boldsymbol{w} \circ \psi \in H(\mathrm{curl}),$$
$$\psi_b^* z = \det(D\psi) z \circ \psi \in L^2.$$

H^1 space construction



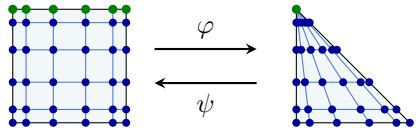
For $v \in \mathcal{Q}_p \subset H^1(\Xi)$, we want $\psi_g^* v \in H^1(\Delta)$.

$$\|\nabla(v \circ \psi)\|^2_{L^2(\Delta)} = \int_\Xi \left(\frac{(\partial_x v)^2}{1-y} + (1-y)\left(\partial_y v + \frac{x\,\partial_x v}{1-y} \right)^2 \right) dx\,dy.$$

For the integral to be finite, $(1-y)$ must divide $\partial_x v$, and so

$$v = (1-y)q + c, \qquad q \in \mathcal{Q}_{p,p-1}, \quad c \text{ constant}.$$

H^1 space construction



For $v \in \mathcal{Q}_p \subset H^1(\Xi)$, we want $\psi_g^* v \in H^1(\Delta)$.

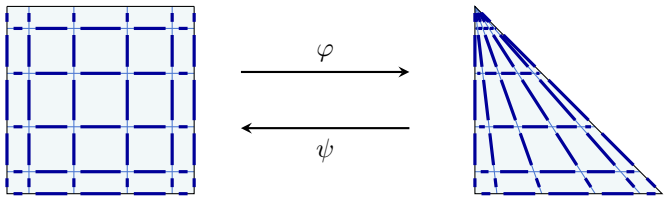
$$\|\nabla(v \circ \psi)\|^2_{L^2(\Delta)} = \int_\Xi \left(\frac{(\partial_x v)^2}{1-y} + (1-y)\left(\partial_y v + \frac{x\,\partial_x v}{1-y} \right)^2 \right) dx\,dy.$$

For the integral to be finite, $(1-y)$ must divide $\partial_x v$, and so

$$v = (1-y)q + c, \qquad q \in \mathcal{Q}_{p,p-1}, \quad c \text{ constant}.$$

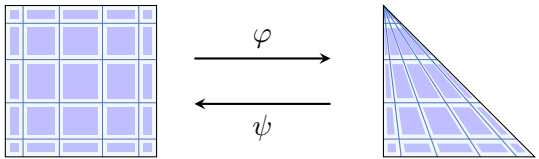
The space is $V = (1-y)\mathcal{Q}_{p,p-1} + \mathcal{Q}_0$. Its **dimension** is $p(p+1)+1$, **exactly the number of vertices in the collapsed lattice**.

$H(\mathrm{curl})$ space construction



- The $H(\mathrm{curl})$ space follows the same principles
- The construction is more subtle
- We have to satisfy both **conformity** and **exactness**
- Arrive at the space $\boldsymbol{W} = \{(t^2 a - t\partial_x b, b) : a \in \mathcal{Q}_{p-1}, b \in \mathcal{Q}_{p,p-1}\}$, where $t = 1 - y$
- Its dimension is $p^2 + p(p+1)$, **exactly the number of edges in the collapsed lattice**.

L^2 space construction



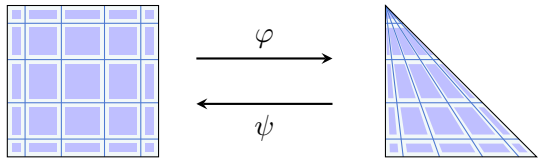
For polynomial z, we require $\psi_b^* z = \det(D\psi) z \circ \psi \in L^2(\Delta)$. Changing variables gives

$$\|\psi_b^* z\|^2_{L^2(\Delta)} = \int_\Xi \frac{z^2}{1-y}\, dx\, dy.$$

For the integral to be finite, $(1-y)$ must divide z and so $z(x,1) \equiv 0$. The space is

$$Z = (1-y)\mathcal{Q}_{p-1}.$$

L^2 space construction



For polynomial z, we require $\psi_b^* z = \det(D\psi) z \circ \psi \in L^2(\Delta)$. Changing variables gives

$$\|\psi_b^* z\|^2_{L^2(\Delta)} = \int_\Xi \frac{z^2}{1-y}\, dx\, dy.$$

For the integral to be finite, $(1-y)$ must divide z and so $z(x,1) \equiv 0$. The space is

$$Z = (1-y)\mathcal{Q}_{p-1}.$$

Its dimension is p^2, **exactly the number of subcells in the collapsed lattice**.

Spaces and their properties

H^1, $H(\mathrm{curl})$ and L^2 conforming finite element spaces on the triangle

$$\mathbb{R} \hookrightarrow \psi_g^* V \xrightarrow{\nabla} \psi_c^* \boldsymbol{W} \xrightarrow{\mathrm{curl}} \psi_b^* Z \longrightarrow 0$$

- The dimension of the spaces equals the number of entities on the collapsed lattice
- The collapsed lattice gives **unisolvent degrees of freedom**
- The spaces form an **exact sequence**
- The high-order spaces contain the **standard** Lagrange, Nédélec (first kind), and L^2 finite elements as subspaces
- The lowest-order spaces **coincide exactly** with the standard ones
- The discrete differential operators are equal to the **signed incidence matrices** of the collapsed lattice connectivity graph
- **Property (P1):** high-order and low-order differentiation matrices are **identical**

Spectral Equivalence

Theorem

Let $A_V, A_{\boldsymbol{W}}, A_Z$ and $A_{V_0}, A_{\boldsymbol{W}_0}, A_{Z_0}$ be the operators induced by the bilinear forms

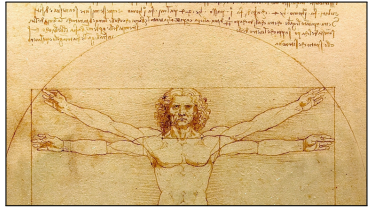
$$\begin{aligned} a_V(u, v) &= (\nabla u, \nabla v), \\ a_{\boldsymbol{W}}(\boldsymbol{u}, \boldsymbol{v}) &= (\nabla \times \boldsymbol{u}, \nabla \times \boldsymbol{v}), \\ a_Z(u, v) &= (u, v). \end{aligned}$$

Then, we have the following spectral equivalences, **independent of** p,

$$\begin{aligned} A_V &\sim A_{V_0}, \\ A_{\boldsymbol{W}} &\sim A_{\boldsymbol{W}_0}, \\ A_Z &\sim A_{Z_0}. \end{aligned}$$

Spectral Equivalence Argument

Let $u_p \in V_p$, $u_h \in V_h$ share the same values at lattice vertices.
Goal: show that $\|\nabla u_p\|_0 \approx \|\nabla u_h\|_0$, independent of p.



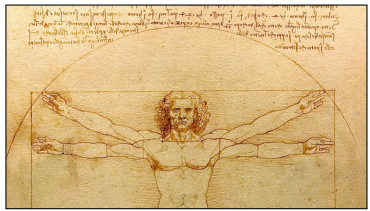
Spectral Equivalence Argument

Let $u_p \in V_p$, $u_h \in V_h$ share the same values at lattice vertices.
Goal: show that $\|\nabla u_p\|_0 \approx \|\nabla u_h\|_0$, independent of p.

Write $\boldsymbol{w}_p = \nabla u_p = \psi_c^* \widetilde{\boldsymbol{w}}_p$. Changing variables,

$$\|\nabla u_p\|_0^2 = \int_\Xi \left\| \begin{pmatrix} (1-y)^{-1} & 0 \\ x(1-y)^{-1} & 1 \end{pmatrix} \begin{pmatrix} \widetilde{w}_{1,p} \\ \widetilde{w}_{2,p} \end{pmatrix} \right\|^2 (1-y)\, dx$$

$$= \int_\Xi \left\| \begin{pmatrix} 1 & 0 \\ x & 1 \end{pmatrix} \begin{pmatrix} (1-y)^{-1}\widetilde{w}_{1,p} \\ \widetilde{w}_{2,p} \end{pmatrix} \right\|^2 (1-y)\, dx.$$



Note that $\left(\begin{smallmatrix} 1 & 0 \\ x & 1 \end{smallmatrix}\right)$ and its inverse have norm bounded by $\varphi = \frac{1+\sqrt{5}}{2}$ for $x \in [0,1]$ **(the golden ratio)**.

So, $$\|\nabla u_p\|_0^2 \approx \left\|(1-y)^{-1/2}\widetilde{w}_{1,p}\right\|_{L^2(\Xi)}^2 + \left\|(1-y)^{1/2}\widetilde{w}_{2,p}\right\|_{L^2(\Xi)}^2.$$

Reduction to 1D

We have shown:

$$\|\nabla u_p\|_0^2 \approx \left\|(1-y)^{-1/2}\widetilde{w}_{1,p}\right\|_{L^2(\Xi)}^2 + \left\|(1-y)^{1/2}\widetilde{w}_{2,p}\right\|_{L^2(\Xi)}^2.$$

Norm equivalences on the triangle can be **reduced** to **weighted** norm equivalences on the square, and in turn to **weighted one-dimensional equivalences**.

- The weights $(1-x)^\alpha(1+x)^\beta$ are known as **Jacobi weights**.
- They give rise to Jacobi–Gauss–Lobatto points and quadrature
  - Standard Gauss–Lobatto corresponds to $\alpha = \beta = 0$.
- Our task is to prove **interpolation stability** and **norm equivalence** in **shifted norms**
  - We use $(\alpha, \beta) = (0, 0)$ for interpolation points, but must prove stability in the $(1, 0)$-norm.

One-Dimensional Equivalences

Let $u_p \in \mathcal{P}_p$ and piecewise linear u_h share Gauss–Lobatto nodal values.

$$\|\partial_x u_p\|_{(\alpha,\beta)} \approx \|\partial_x u_h\|_{(\alpha,\beta)}, \qquad \alpha, \beta \in \{0, 1\}.$$
$$\|u_p\|_{(\alpha,\beta)} \approx \|u_h\|_{(\alpha,\beta)}, \qquad u_p \in \mathcal{P}_{p-1}.$$

We obtain the following estimates, all independent of p.

Interpolation Equivalences

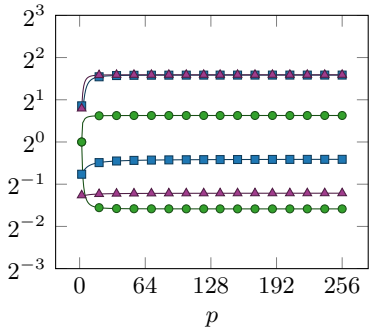
$$\|u_p\|_0 \approx \|u_h\|_0,$$
$$\|\partial_x u_p\|_{(\alpha,0)} \approx \|\partial_x u_h\|_{(\alpha,0)}, \qquad \alpha \in \{0, 1\},$$
$$\|u_p\|_{(-1,0)} \approx \|u_h\|_{(-1,0)}, \qquad u_p(1) = 0,$$
$$\|u_p\|_{(1,0)} \approx \|u_h\|_{(1,0)}, \qquad u_p \in \mathcal{P}_{p-1}.$$

Histopolation Equivalence

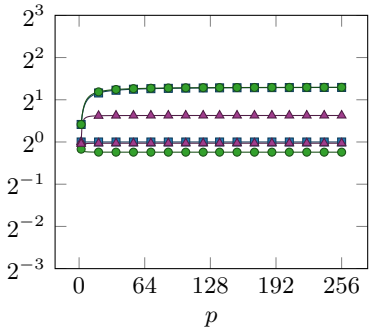
$$\|\widehat{u}_p\|_{(\alpha,0)} \approx \|\widehat{u}_h\|_{(\alpha,0)}, \quad \alpha \in \{0, 1\},$$
$$\|\widehat{z}_p\|_{(-1,0)} \approx \|\widehat{z}_h\|_{(-1,0)}.$$

Computed 1D Bounds

Interpolation Norm Equivalences



Histopolation Norm Equivalences



2D Bounds

- The 1D bounds can be combined to give upper and lower 2D bounds

$$c\|\nabla u_h\|_0^2 \leq \|\nabla u_p\|_0^2 \leq C\|\nabla u_h\|_0^2$$

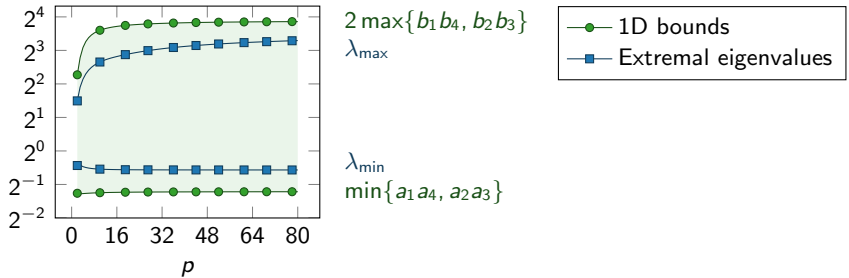
Lower Bound

$$c = \varphi^{-4} \min\{a_1 a_4, a_2 a_3\}$$

Upper Bound

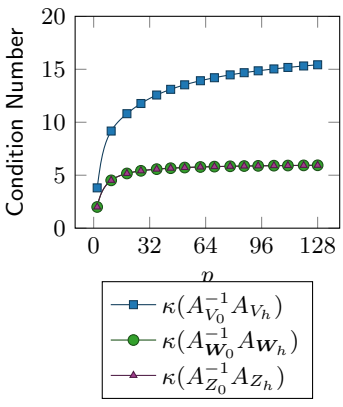
$$C = \varphi^{4} \max\{b_1 b_4, b_2 b_3\}$$

Factors of φ^4 are pessimistic. Shown below: $\min\{a_1 a_4, a_2 a_3\}$ and $2\max\{b_1 b_4, b_2 b_3\}$.

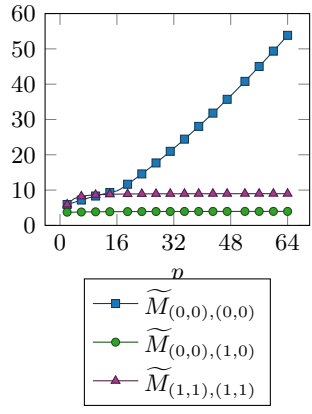


Condition Numbers

Condition Number on the Reference Triangle

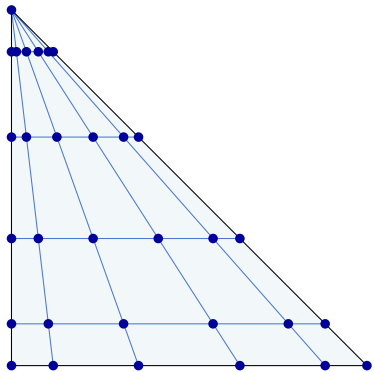


Diagonally Scaled Mass Matrix



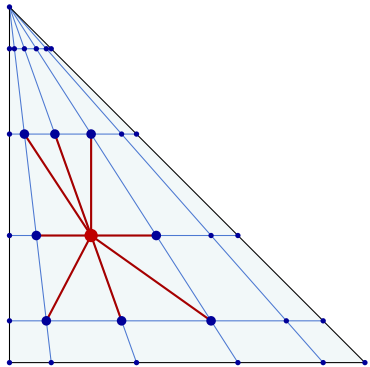
Sparsity and AMG

- For high-order operator, $\text{nnz}(A)$ per row is $\mathcal{O}(p^2)$
- We saw average $\text{nnz}(A_0)$ per row ≤ 9 for all p and h



Sparsity and AMG

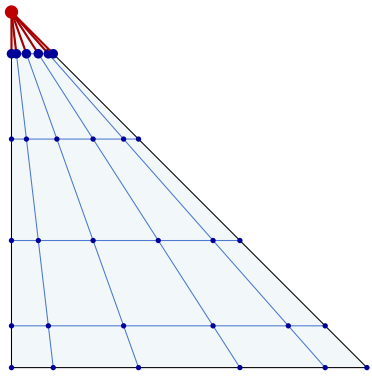
- For high-order operator, $\mathrm{nnz}(A)$ per row is $\mathcal{O}(p^2)$
- We saw average $\mathrm{nnz}(A_0)$ per row ≤ 9 for all p and h
- Except for the top vertex, each low-order vertex is coupled to **at most 9** vertices, including itself



9 couplings, including itself

Sparsity and AMG

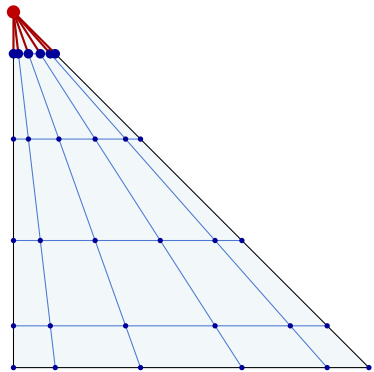
- For high-order operator, $\mathrm{nnz}(A)$ per row is $\mathcal{O}(p^2)$
- We saw average $\mathrm{nnz}(A_0)$ per row ≤ 9 for all p and h
- Except for the top vertex, each low-order vertex is coupled to **at most 9** vertices, including itself
- The top vertex is coupled to its $\boldsymbol{p+1}$ adjacent vertices



$p+1$ adjacent vertices

Sparsity and AMG

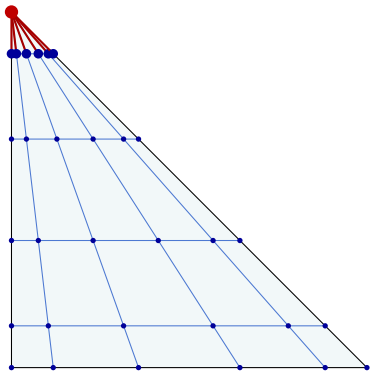
- For high-order operator, $\text{nnz}(A)$ per row is $\mathcal{O}(p^2)$
- We saw average $\text{nnz}(A_0)$ per row ≤ 9 for all p and h
- Except for the top vertex, each low-order vertex is coupled to **at most 9** vertices, including itself
- The top vertex is coupled to its $\boldsymbol{p+1}$ adjacent vertices
- Total of $\boldsymbol{9p^2 - p + 1}$ nonzeros in element matrix with $p^2 + p + 1$ rows
- $\implies$ average of **9** nonzeros per row as $p \to \infty$



$p + 1$ adjacent vertices

Sparsity and AMG

- For high-order operator, $\text{nnz}(A)$ per row is $\mathcal{O}(p^2)$
- We saw average $\text{nnz}(A_0)$ per row ≤ 9 for all p and h
- Except for the top vertex, each low-order vertex is coupled to **at most 9** vertices, including itself
- The top vertex is coupled to its $\boldsymbol{p+1}$ adjacent vertices
- Total of $\boldsymbol{9p^2 - p + 1}$ nonzeros in element matrix with $p^2 + p + 1$ rows
- $\implies$ average of **9** nonzeros per row as $p \to \infty$

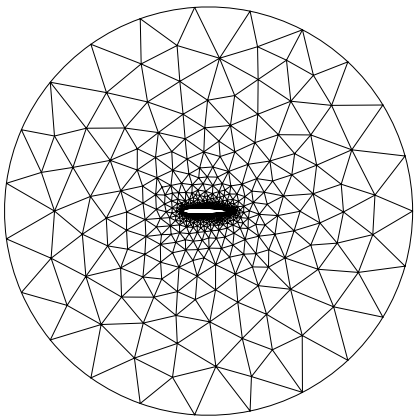


$p + 1$ adjacent vertices

AMG: A_0 has non-standard structure, can be challenging for AMG. We use two sweeps of symmetric ℓ^1 hybrid Gauss-Seidel and no aggressive coarsening.

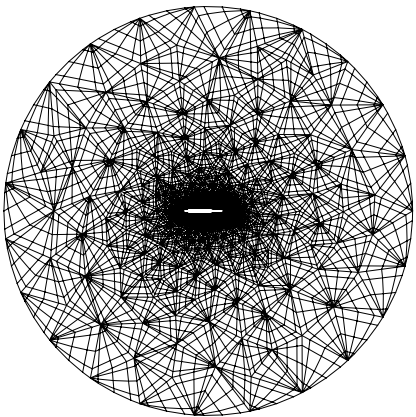
Numerical Examples: NACA Mesh

$$(u, v) + (\nabla u, \nabla v) = (f, v), \qquad u = 0 \text{ on } \partial\Omega$$



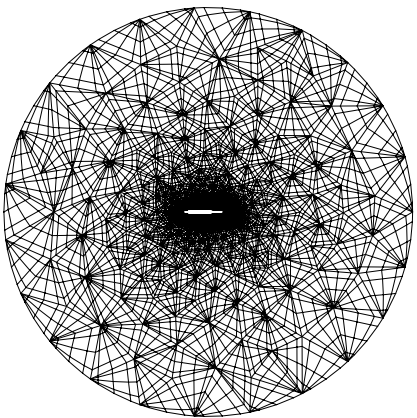
Numerical Examples: NACA Mesh

$$(u, v) + (\nabla u, \nabla v) = (f, v), \qquad u = 0 \text{ on } \partial\Omega$$



Numerical Examples: NACA Mesh

$$(u, v) + (\nabla u, \nabla v) = (f, v), \qquad u = 0 \text{ on } \partial\Omega$$



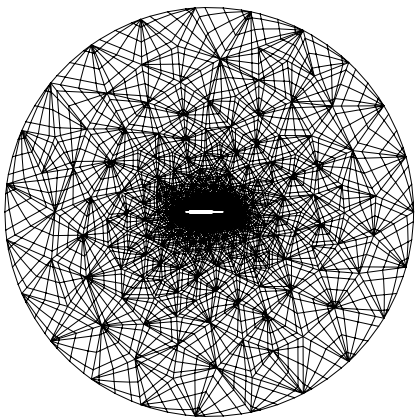
p refinement

p	# It.	# DOFs	$\mathrm{nnz}(A)$ per row	$\mathrm{nnz}(A_0)$ per row
2	18/18	5,766	11.68	8.12
4	23/27	26,300	28.46	8.62
8	25/33	111,672	86.17	8.82
16	30/40	459,632	297.66	8.91
32	38/52	1,864,416	1,104.66	8.96

Number of iterations: direct(A_0) / AMG(A_0)

Numerical Examples: NACA Mesh

$$(u, v) + (\nabla u, \nabla v) = (f, v), \qquad u = 0 \text{ on } \partial\Omega$$



h refinement ($p = 8$)

Ref.	# It.	# DOFs	nnz(A) per row	nnz(A_0) per row
0	25/33	111,672	86.17	8.82
1	27/36	444,864	86.49	8.84
2	28/40	1,775,808	86.64	8.85
3	28/44	7,095,936	86.72	8.86
4	27/47	28,369,152	86.76	8.86

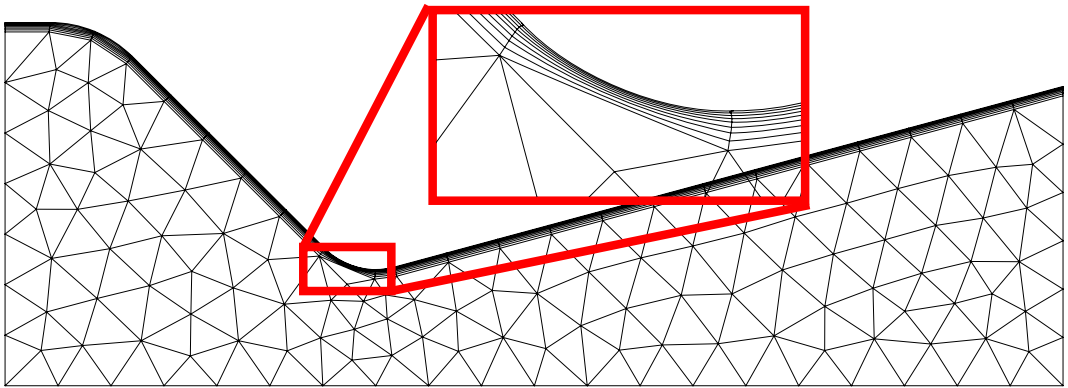
Number of iterations: direct(A_0) / AMG(A_0)

Sum Factorization

- The new spaces are constructed using pull backs from the triangle to the square
- On the square, the spaces have a natural tensor-product structure that enabled **sum factorization**
- The H^1 **reference square space** can be evaluated using **the same kernels** as **standard $\mathcal{Q}_p$ Lagrange elements on quadrilaterals**
- Complexity of operator evaluation **reduced** from $\mathcal{O}(p^4)$ to $\mathcal{O}(p^3)$
- Using matrix-free LOR, can **avoid assembly**, eliminating $\mathcal{O}(p^6)$ complexity

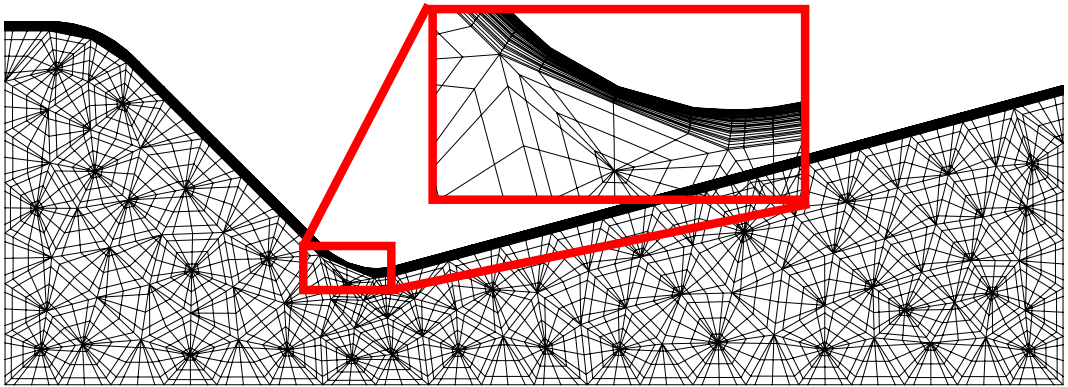
Mixed Meshes

- Gauss–Lobatto lattice points means the triangular elements are **compatible with standard Gauss–Lobatto quadrilateral elements**
- Can easily handle mixed meshes of triangles and quadrilaterals
- **Example:** BGM45-15 nozzle with boundary layer



Mixed Meshes

- Gauss–Lobatto lattice points means the triangular elements are **compatible with standard Gauss–Lobatto quadrilateral elements**
- Can easily handle mixed meshes of triangles and quadrilaterals
- **Example:** BGM45-15 nozzle with boundary layer



Mixed Meshes

- Gauss–Lobatto lattice points means the triangular elements are **compatible with standard Gauss–Lobatto quadrilateral elements**
- Can easily handle mixed meshes of triangles and quadrilaterals
- **Example:** BGM45-15 nozzle with boundary layer

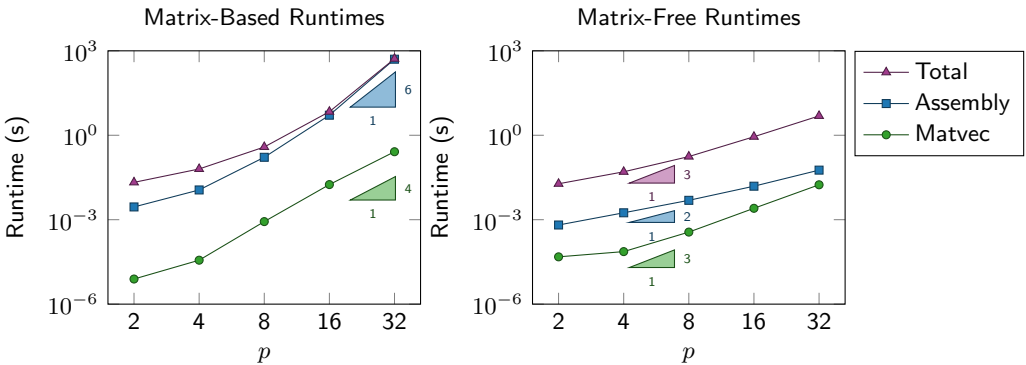
BGM45-15 nozzle: p refinement

p	# It.	# DOFs	$\mathrm{nnz}(A)$ per row	$\mathrm{nnz}(A_0)$ per row
2	25/25	1,481	13.59	8.40
4	27/30	6,225	31.70	8.72
8	30/33	25,505	92.02	8.86
16	34/39	103,233	308.71	8.93
32	39/52	415,361	1,126.12	8.97

Number of iterations: direct(A_0) / AMG(A_0)

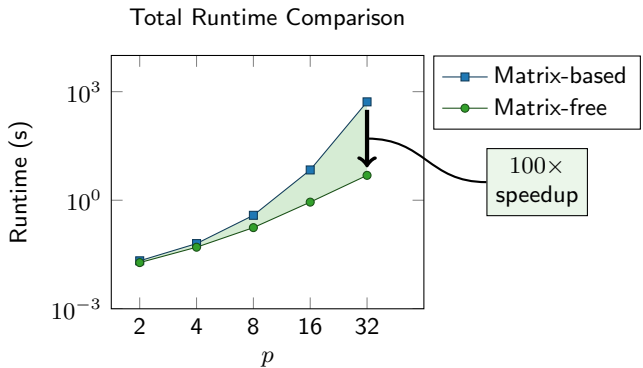
Runtime Comparison

Implemented in MFEM, using existing partial assembly factorization kernels



Runtime Comparison

Implemented in MFEM, using existing partial assembly factorization kernels



Fictitious Space Preconditioner

Fictitious Space Lemma (Nepomnyaschikh)

$A : V \to V$, $\widehat{A} : \widehat{V} \to \widehat{V}$, $R : \widehat{V} \to V$ surjective, $S : V \to \widehat{V}$ right-inverse, and

$$\|R\widehat{v}\|_A^2 \leq c_R \|\widehat{v}\|_{\widehat{A}}^2, \qquad \|Sv\|_{\widehat{A}}^2 \leq c_S \|v\|_A^2.$$

Define the preconditioner $B = R\widehat{A}^{-1}R^T$. The condition number satisfies

$$\kappa(BA) \leq c_R c_S.$$

Fictitious Space Preconditioner

Fictitious Space Lemma (Nepomnyaschikh)

$A : V \to V$, $\widehat{A} : \widehat{V} \to \widehat{V}$, $R : \widehat{V} \to V$ surjective, $S : V \to \widehat{V}$ right-inverse, and

$$\|R\widehat{v}\|_A^2 \leq c_R \|\widehat{v}\|_{\widehat{A}}^2, \qquad \|Sv\|_{\widehat{A}}^2 \leq c_S \|v\|_A^2.$$

Define the preconditioner $B = R\widehat{A}^{-1}R^T$. The condition number satisfies

$$\kappa(BA) \leq c_R c_S.$$

- Use the **Duffy space** as the fictitious/auxiliary space for the **Lagrange space**
- Set $R : \widehat{V} \to V$ to be local elliptic projection $\implies c_R \lesssim 1$
- Right inverse $S : V \hookrightarrow \widehat{V}$ inclusion $\implies c_S = 1$
- $\implies \kappa(BA) = \mathcal{O}(1)$

Fictitious Space Results

NACA0012

p	# It. B	# It. B_0	# DOFs	Block size
2	13/18	15/16	3,920	0
4	13/20	19/23	15,224	3
8	13/22	20/28	59,984	21
16	13/27	20/32	238,112	105
32	14/27	19/35	948,800	465

BGM45-15 nozzle

p	# It. B	# It. B_0	# DOFs	Block size
2	18/20	31/32	1,257	0
4	16/19	25/28	4,881	3
8	15/19	28/32	19,233	21
16	15/26	29/34	76,353	105
32	16/28	30/37	304,257	465

B: high-order Duffy fictitious space; B_0: low-order refined fictitious space.

Number of iterations: direct / AMG for the fictitious system.

Conclusions

- Construction of finite element de Rham complex on triangles
- High-order and low-order discrete differential operators are **identical**
  - Signed incidence matrix of lattice connectivity
- Spectral equivalence, **independent of** p, with low-order discretization
  - Reduced to sequence of shifted Jacobi-weighted 1D equivalences
- Amenable to **sum factorization** and **matrix-free methods**
- Precondition the standard finite element spaces using a fictitious space

Thank you!



