On the Impact of Triangle Shapes for Boundary Layer Problems Using High-Order Finite Element Discretization

Huafei Sun, David L. Darmofal, and Robert Haimes

Aerospace Computational Design Laboratory, Massachusetts Institute of Technology

The impact of triangle shapes, including angle sizes and aspect ratios, on accuracy and stiffness is investigated for simulations of highly anisotropic problems. The results indicate that for high-order discretizations, large angles do not have an adverse impact on solution accuracy. However, a correct aspect ratio is critical for accuracy for both linear and high-order discretizations. Large angles are also found to be not problematic for the conditioning of the linear systems arising from the discretizations. Further, when choosing preconditioning strategies, coupling strengths among elements rather than element angle sizes should be taken into account. With an appropriate preconditioner, solutions on meshes with and without large angles can be achieved within a comparable time.

I. Introduction

Many physical problems exhibit highly anisotropic behaviors, where the length scale in one direction can be several orders of magnitude smaller than in the other. To resolve these features, which often occur in thin boundary layers, wakes, or shocks, highly anisotropic elements need to be employed to improve computational efficiency. In turbulence applications, aspect ratios on the order of 100,000 are often appropriate. Therefore, generating highly anisotropic meshes during adaptation becomes a key step in the efficient simulation of such applications.

Understanding the impact of triangle shapes provides an important guideline for mesh generation and adaptation. While high aspect ratio meshes are important for efficiency to resolve anisotropic features, triangular elements with large angles are known to deteriorate solution quality. Babuška and Aziz demonstrated that the accuracy of a finite element approximation on triangular meshes degrades as the maximum angle of elements approaches $180^\circ$ asymptotically. Although little can be said about elements of intermediate quality in a non-asymptotic range, avoiding large angles leads to the development of several mesh generation algorithms, including minimum-maximum triangulations, and hybrid methods with structured meshes in the regions where high anisotropy is needed. Both of these methods have different robustness and automation problems, especially for complex geometries, and avoiding large angles in mesh generation still remains a difficult problem.

Shewchuk conducted a comprehensive study on triangle shapes for linear finite elements. He concluded that the triangle shape that leads to the best accuracy in terms of the $H^1$-seminorm of interpolation error is one with an aspect ratio equal to the ratio $\kappa$ of the maximum and minimum singular values of the solution Hessian but without large angles. However, he also demonstrated that this so-called “superaccuracy” is very fragile as the elements have to align precisely with the singular vectors of the Hessian, and so he suggested that generating triangles with an intermediate aspect ratio of $\sqrt{\kappa}$ but possibly with large angles is a more realistic goal for anisotropic mesh generation. Also, Rippa showed in his work that this aspect ratio is optimal for the $L^p$-norm of the interpolation error for $1 \leq p < \infty$. Although these results provide good insight into what an optimal triangle shape should be, their analysis is limited to linear elements, and to authors’ knowledge, the impact of large angles and aspect ratios has not been studied for high-order discretizations.
of highly anisotropic problems. If high-order discretizations are not sensitive to element angles, then large
groups of highly anisotropic problems. If high-order discretizations are not sensitive to element angles, then large
angles introduced by a Delaunay triangulation in a stretched space \(^8,9\) would not be problematic.

Another important aspect of numerical simulation is the solution efficiency of the linear system arising from
the discretization applied. Many have investigated the relationship between mesh quality (e.g. element
angle size) and the condition number of the stiffness matrix arising from a standard Galerkin approach for a self-adjoint elliptic problem. For example, Shewchuk derived a bound on the stiffness matrix condition
number for standard linear finite element, \(^6\) and recently in the work of Du et al., \(^10\) a refined bound was derived
for a high-order finite element. However, little can be inferred from their work when convection terms are
present, which make the stiffness matrix non-symmetric. Further, although stiffness matrix conditioning is
important for the efficiency of iterative linear solvers, investigating preconditioning strategies for meshes with
different geometric properties has a more practical value since for many applications, only the preconditioned
systems are tractable.

In this paper, the impact of large angles and element aspect ratios will be studied for boundary layer
problems using high-order discretizations. Both solution accuracy and conditioning of the discretized system
will be investigated when large angles are present. Results show that using a high-order discretization, the
same accuracy can be achieved at similar degrees of freedom for both large-angle and right-angle triang-
ulations. However, a correct aspect ratio is critical for accuracy. It is also found that the two mentioned
triangulations have similar linear conditioning, but solution schemes for these two should be different as they
involve different couplings among elements. Incomplete LU factorization, being a popular preconditioning
scheme, is investigated for effectiveness on both triangulations.

This paper is organized as follows. Section II describes the discretization and the solution scheme applied.
Sections III and IV present the model problems and the meshes considered, including large-angle elements
and right-angle elements. Section V studies the discretization errors on these meshes, and Section VI studies
conditioning of the linear systems arising from these meshes.

II. Discretization and Solution Method

II.A. Discontinuous Galerkin Discretization

Let \( \Omega \in \mathbb{R}^d \) be an open, bounded domain in a \( d \)-dimensional space. A general steady state conservation law
in the domain, \( \Omega \), expressed in the strong form is given by

\[
\nabla \cdot F^i(u) - \nabla \cdot F^v(u, \nabla u) = S(x), \quad \text{in } \Omega, \tag{1}
\]

where \( u(x) : \mathbb{R}^d \rightarrow \mathbb{R}^m \) is the \( m \)-state solution vector. The inviscid flux \( F^i(u) : \mathbb{R}^m \rightarrow \mathbb{R}^{m \times d} \), the viscous
flux \( F^v(u, \nabla u) : \mathbb{R}^m \times \mathbb{R}^{m \times d} \rightarrow \mathbb{R}^{m \times d} \), and the source term \( S(x) : \mathbb{R}^d \rightarrow \mathbb{R}^m \) characterize the governing
equations to be solved.

The discretization of (1) is carried out using a Discontinuous Galerkin (DG) method on triangular meshes.
Let \( T_H \) be a triangulation of the domain \( \Omega \) with non-overlapping elements, \( \kappa \), such that \( \Omega = \bigcup_{\kappa \in T_H} \bar{\kappa} \). Also,
define a function space \( V^p_H \) by

\[
V^p_H \equiv \{ v \in (L^2(\Omega))^m : v|_\kappa \in (P^p(\kappa))^m, \forall \kappa \in T_H \}, \tag{2}
\]

where \( P^p(\kappa) \) denotes the space of the \( p \)-th order polynomials on \( \kappa \). Multiplying (1) by a test function,
\( v_H \in V^p_H \), and integrating by parts over all elements leads to the weak formulation of the conservation law,
which reads as follows: find \( u_H(\cdot) \in V^p_H \) such that

\[
R_H(u_H, v_H) = 0, \quad \forall v_H \in V^p_H, \tag{3}
\]

where the semi-linear weighted residual (linear in the second argument) is given by

\[
R_H(u_H, v_H) = \sum_{\kappa} \left[ E_\kappa(u_H, v_H) + V_\kappa(u_H, v_H) + S_\kappa(v_H) \right],
\]

and \( E_\kappa(u_H, v_H) \), \( V_\kappa(u_H, v_H) \), and \( S_\kappa(v_H) \) denote the contributions of the inviscid, viscous, and source terms,
respectively. Specifically, Roe numerical flux function \(^{11}\) is used for the inviscid discretization, \( E_\kappa(u_H, v_H) \).
The viscous flux contribution, \( V_\kappa(u_H, v_H) \), is discretized according to the second form of Bassi and Rebay
(BR2), \(^{12}\) which is dual consistent and requires only the nearest neighbor coupling. Further details on the
discretization of the viscous term can be found in Fidkowski et al.\textsuperscript{13} The discretization of the source term is given by $S_a(v_H) \equiv - \int_{\Omega_e} v_H^T S(x) dx$. Boundary conditions are enforced weakly by appropriately setting the numerical flux on the domain boundaries. The boundary treatment for the Navier-Stokes equations can be found in Oliver.\textsuperscript{14}

To obtain the discrete form of the governing equation, a basis for the function space, $\mathcal{V}^p_\mathcal{H}$, is chosen. Denote this basis by $\{ \phi_i \}$ for $i = 1, \ldots, N$, and let $U_H \in \mathbb{R}^N$ be the vector of expansion coefficients for the solution $u_H \in \mathcal{V}^p_\mathcal{H}$, that is, $u_H = \sum_{i=1}^{N} U_H, \phi_i(x)$. Then the discrete form of the weak formulation in (3) can be written as a system of algebraic equations: find $U_H$ such that

$$R_H(U_H) = 0,$$ \hspace{1cm} (4)

where $R_H(U_H)$ is the discrete residual vector such that $R_H(U_H)_{i} = R_H(u_H, \phi_i)$.

### II.B. Linear Solution Method

The system defined in (4) is solved using Newton’s method, where the solution vector at the $n+1$st iteration is given by

$$U_H^{n+1} = U_H^n - \left( \frac{\partial R_H}{\partial U_H}_\bigg|_{U_H^n} \right)^{-1} R_H(U_H^n).$$

This requires at each iteration the solution to the linear system $Kx = b$, where

$$K \equiv \frac{\partial R_H}{\partial U_H}_\bigg|_{U_H^n}, \hspace{1cm} x \equiv \Delta U_H^n, \hspace{1cm} b \equiv -R_H(U_H^n),$$ \hspace{1cm} (5)

The matrix $K$ is referred to as the Jacobian matrix. In the implementation of solving (4), a pseudo-time stepping is applied in order to improve the solver robustness. The time step is raised to essentially infinite value as the solution converges to steady state, at which the linear system involved is the same as (5). For the DG discretization presented, the Jacobian matrix has a sparse block-structure with $N_e$ block rows, where $N_e$ is the number of elements in the computational domain. Each block has a size of $n_b \times n_b$, where $n_b$ is the number of degrees of freedom in one element. Each block row has a non-zero diagonal block, which corresponds to the coupling between degrees of freedom within one element. In addition, because the BR2 scheme requires only the nearest neighbor coupling,\textsuperscript{12} each block row of the Jacobian matrix also involves $n_f$ off-diagonal blocks, where $n_f$ is the number of faces per element ($n_f = 3$ for 2D).

Knowing the large size of the Jacobian matrix, iterative methods are used to solve the linear system given in (5). More specifically, a restarted GMRES algorithm is applied.\textsuperscript{15,16} To expedite the convergence of the GMRES algorithm, a preconditioner must be applied in general to improve the conditioning of the linear system. In this work, the preconditioner considered is a dual threshold incomplete LU factorization,\textsuperscript{17} ILUT($p, \tau$), where $p$ is the number of allowed fill-in’s per row, and $\tau$ is the fill-in drop threshold. Since the Jacobian matrix has a block structure, ILUT($p, \tau$) is implemented in a block form, where the Frobenius norm is used to measure the contribution of a block. The standard ILU($0$), being a particular case of ILUT($p, \tau$) with $p = 0$ and $\tau \rightarrow \infty$, is also considered in this study and implemented in a block form.

The efficiency of incomplete factorization is highly dependent on the ordering of unknowns, especially for a non-symmetric system.\textsuperscript{18} The reordering scheme used in this work is the Minimum Discarded Fill (MDF) method, where the element that produces the least discarded fill-in is ordered first and the process is repeated in a greedy manner. This method was presented by D’Azevdo et al.,\textsuperscript{19} and modified for a block matrix and reported to work well with ILU($0$) in a DG context.\textsuperscript{20} For the block form of the MDF algorithm, each block of the matrix is reduced to a scalar using the Frobenius norm, and the reordering is based on the reduced scalar matrix.

### III. Model Problems

To study the impact of large angles and aspect ratios, a thin boundary layer case is used for this study, and the governing equations considered include the advection-diffusion equation and the Navier-Stokes equations, both of which have the form of the conservation law given in (1).
III.A. Advection-Diffusion Equation

The advection-diffusion equation has its inviscid and viscous fluxes defined as $F^i \equiv \vec{\beta}u$ and $F^v \equiv \bar{\mu} \cdot \nabla u$, respectively, where $\vec{\beta}$ is the advection velocity and $\bar{\mu}$ is the diffusivity tensor. When $\bar{\mu}$ is isotropic, its notation is replaced by a scalar $\mu$.

For the anisotropic problem considered in this work, the convection velocity is uniform and horizontal with a unit magnitude, and a diffusivity of $\mu = 10^{-8}$ is introduced. The computational domain is a rectangular box of $[0.05, 1.05] \times [0, 0.001]$. A source term is added such that the exact solution to this problem has a form of

$$u = 1 - e^{-\sqrt{c}x},$$

with $c = 0.59$. This solution is shown in Figure 1 and resembles a thin boundary layer growing with $\sqrt{x}$ along the bottom wall. It has a thickness of $\delta_{0.99} = 8 \times 10^{-5}$ at the inflow and $3.6 \times 10^{-4}$ at the outflow. Note that the leading edge of the boundary layer is not included in the computational domain.

![Figure 1. Exact solution of $u$ given in (6)](image)

III.B. Navier-Stokes Equations

For the compressible Navier-Stokes equations, the conservative state vector is $u = [\rho, \rho\vec{v}, \rho E]_T$, where $\rho$ is density, $\vec{v}$ is velocity, and $E$ is specific total internal energy. The inviscid and viscous fluxes, $F^i$ and $F^v$, are given such that (1) represents a compact notation for the conservation of mass, momentum, and energy. In addition, the ideal gas law is assumed. A source term is created to make an exact solution of $\rho, P = \text{const}, v = 0$, and $u$ given by (6), with $Re = 10^8$ and $M_\infty = 0.1$. The computational domain is the same as for the case of the advection-diffusion equation.

IV. Meshes

As previously cited, a realistic goal for anisotropic meshing for linear elements is to generate triangles with an aspect ratio of $\sqrt{\sigma_{\text{max}}/\sigma_{\text{min}}}$, where the $\sigma$’s are the singular values of the solution Hessian. Also, at this aspect ratio, angle size is not critical for solution accuracy. In fact, this value is the target aspect ratio in many anisotropic adaptation schemes for linear discretizations, in which interpolation errors are equidistributed in all directions. In this paper, this aspect ratio is used as a guideline for the meshes considered, and the element orientation is decided based on the angle $\theta$ of the singular vectors of the Hessian matrix. Further, for high-order schemes to achieve equidistribution of interpolation errors, Fidkowski proposed an aspect ratio given by

$$AR = \left(\frac{u_{e_1}^{(p+1)}}{u_{e_2}^{(p+1)}}\right)^{1/(p+1)},$$

(7)
where \( p \) is the interpolation order, \( u^{(p+1)} \) represents the \( p + 1 \)st derivative of \( u \) along the direction \( e \), \( e_1 \) is the direction of maximum \( p + 1 \)st derivative, and \( e_2 \), the direction orthogonal to \( e_1 \).

Figure 2 shows the ratio \( \sigma_{\text{max}}/\sigma_{\text{min}} \) of the Hessian of the solution considered in (6). Along the bottom wall, which will be the main source of discretization error, the ratio of \( \sigma_{\text{max}}/\sigma_{\text{min}} \) is about \( 10^8 \), so an aspect ratio of \( AR \approx 10^4 \) might be expected to result in a high-quality solution for linear elements. Three different aspect ratios are considered in this work: \( 10^3 \), \( 10^4 \), and \( 10^5 \). Also, for the solution considered, \( \theta \) is essentially zero everywhere in the computational domain, and has a maximum value of \( 0.14^\circ \). This indicates that the major axes of elements should be aligned horizontally. In addition, Figure 3 shows the ratio \( u^{(4)}_{e_1}/u^{(4)}_{e_2} \), where \( e_1 \) and \( e_2 \) are defined as in (7). As shown, this ratio is about \( 10^{16} \) along the bottom wall. This indicates that equidistribution of interpolation error in all directions leads to an aspect ratio of about \( AR \approx 10^4 \) for an interpolation order of \( p = 3 \) according to (7).

To study the impact of large angles, elements with angles close to \( 180^\circ \) (obtuse triangles) are compared with right-angle elements (acute triangles). Figure 4 shows the meshes of these two kinds of triangles. Note the \( y \)-axis is rescaled only for the convenience of visualization, so in the physical domain, each of the obtuse triangles has one angle of about \( 180^\circ \), and each of the acute triangles has essentially two angles of about \( 90^\circ \). For the aspect ratio of \( 10^4 \), the large angle in the obtuse triangles is \( 179.9771^\circ \).
V. Discretization Error

An error convergence study is conducted for the advection-diffusion equation and the Navier-Stokes equations on the meshes of all aspect ratios considered. The $L^2$ error and the error of a selected output are used to assess the solution quality on these meshes. In engineering applications, the error of a functional output often draws more interests than the global error. For the advection-diffusion problem, the error of the heat flux across the bottom wall is measured. This output is given by

$$J = \int_{\text{bottom wall}} \mu \frac{\partial u}{\partial y} \, dx.$$  

(8)

For the Navier-Stokes equations, the chosen output of interest is the drag along the bottom wall, which is also given by (8) for the model problem considered.

In this section, the $L^2$ errors are normalized by the $L^2$ norm of the true solution, that is,

$$E \equiv \frac{\|u - u_e\|_{L^2(\Omega)}}{\|u_e\|_{L^2(\Omega)}},$$

where $u_e$ is the exact solution. The output errors are normalized by the true output value. For all cases, solution orders of $p = 1$ and $p = 3$ are considered.

V.A. Advection-Diffusion Equation

The normalized $L^2$ errors for the acute and obtuse meshes of different aspect ratios are shown in Figure 5, where $N$ is the total number of elements. For the linear solutions, an aspect ratio of $\mathcal{R} = 10^4$ generally leads to the lowest error at a given DOF for both shapes. This result is consistent with the conclusion that an aspect ratio of $\sqrt{\sigma_{\text{max}}/\sigma_{\text{min}}}$ leads to the lowest $L^2$ error for linear elements regardless of angle sizes.\(^6,7,25\)

While the solution quality degrades slightly when the aspect ratio changes from $10^4$ to $10^3$, the degradation is more severe for $\mathcal{R} = 10^5$. In addition, the adverse impact of large angles is apparent only for $\mathcal{R} = 10^5$. For the high-order solutions, the aspect ratios of $10^4$ and $10^3$ lead to very similar errors, but $\mathcal{R} = 10^5$ results in a significantly higher error. However, the difference between the acute and obtuse meshes is minor for any aspect ratio. In other words, while the impact of aspect ratios on solution accuracy is critical, the impact of large angles is minor for high-order schemes.

Figure 5 shows the normalized heat flux error for the acute and obtuse meshes. The results lead to a similar conclusion to what was observed for the $L^2$ error, where only the linear solutions are susceptible to large angles; that is, the difference between the two element shapes is less pronounced for high-order solutions than linear solutions. However, a correct aspect ratio is critical for the error level. While the best
aspect ratio is $AR = 10^4$ for both element shapes and for both interpolation orders, one important difference compared to the results of $L^2$ error is that $AR = 10^4$ is considerably better than both $AR = 10^3$ and $10^5$ for linear elements. Also, note that the error reaches machine precision as the meshes are refined for the high-order solutions.

One question that naturally arises is whether these results are only observed under the DG discretization applied. Consider the $L^2$ projection of the exact solution, denoted by $u_{L^2}$, and denote the DG solution by $u_{DG}$. Already seen from Figure 5 for the DG solution $u_{DG}$, the difference between the acute and obtuse triangles is pronounced at this aspect ratio, especially for $p = 1$. In contrast, this difference is minor for $u_{L^2}$ for both $p = 1$ and $p = 3$, as shown in Figure 7, which compares the $L^2$ error of $u_{L^2}$ and $u_{DG}$ at $AR = 10^5$. In fact, it is found that $u_{L^2}$ is not sensitive to element angle size for all the aspect ratios considered. This indicates the possibility of designing a discretization scheme that is insensitive to element angle size, for example, a scheme with its energy norm close to $L^2$ norm. However, energy norm being the same as $L^2$ norm is not enough for an accurate computation of the output, which has a more practical value than $L^2$ error. Figure 8 compares the heat flux error of $u_{L^2}$ and $u_{DG}$ at $AR = 10^5$. Although $u_{L^2}$ is not susceptible to large angles, it leads to a significantly larger heat error than $u_{DG}$.

Figure 6. Normalized heat flux error for the advection-diffusion problem

![Figure 6](image)

(a) Solution Order of $p = 1$

(b) Solution Order of $p = 3$

Figure 7. Normalized $L^2$ error of $L^2$ projection of exact solution and DG solution at $AR = 10^5$

![Figure 7](image)

(a) Solution Order of $p = 1$

(b) Solution Order of $p = 3
V.B. Navier-Stokes Equations

Figure 9 shows the convergence of the normalized $L^2$ error of $\rho u$ for the Navier-Stokes equations. The conclusion drawn is very similar to that of the $L^2$ error for the advection-diffusion case. For the linear solutions, an aspect ratio of $\mathcal{R} = 10^4$ generally leads to the lowest error at a given DOF for both element shapes, and the degradation of solution quality is more severe for $\mathcal{R} = 10^5$ than for $\mathcal{R} = 10^3$. Also, the impact of large angles is more pronounced for $\mathcal{R} = 10^5$ than other aspect ratios. For the high-order solutions, the aspect ratios of $10^4$ and $10^3$ lead to very similar errors, but $\mathcal{R} = 10^5$ results in a significantly higher error. However, the difference between the two triangle shapes is minor for any aspect ratio.

Figure 10 shows the normalized output error for the acute and obtuse meshes, where the output is selected to be drag on the bottom wall. For linear solutions, at the aspect ratios of $10^3$ and $10^4$, the obtuse and acute elements lead to similar error levels. However, at $\mathcal{R} = 10^5$, the acute elements outperform the obtuse ones significantly, and in fact, this shape leads to the lowest error for linear solutions. This optimality of acute triangles is very fragile in the sense that the error significantly increases when the aspect ratio deviates from $10^5$. In fact, as it decreases to $10^3$, the acute element becomes the shape with the highest error for linear solutions. While this result seems to agree with Shewchuk’s conclusion about the fragile
superaccuracy” of the $H^1$-seminorm of interpolation error for acute linear elements, the same behavior is not observed for the heat error in the advection-diffusion problem as in Figure 6. Interpolation theory seems insufficient to explain this difference between output errors of two different governing equations under the DG discretization applied. On the other hand, for high-order solutions in Figure 10, the obtuse elements lead to a similar error compared to the acute ones for any aspect ratio, and the best triangle shape is the obtuse ones with $\mathcal{A}^R = 10^4$. Furthermore, for drag error, the impact of aspect ratio is again very pronounced for both linear and high-order solutions and for both element shapes.

![Figure 10. Normalized drag error for the Navier-Stokes equations](image)

Not only have the high-order solutions been shown to be less sensitive to element angles, the use of high-order schemes can also be justified from the perspective of the degrees of freedom needed for an error level. Figure 11 presents the worst and best output errors from the linear solutions and the high-order solutions from Figure 10. Clearly, the high-order solutions lead to a lower error for almost all degrees of freedom. To put the error of drag in context, consider for example a typical large, long-range, passenger jet, an error of 1% for drag translates into approximately 4-8 passengers, depending on whether the configuration is limited by fuel volume or weight. Thus, for the two-dimensional boundary layer problem studied, an output error level of $0.01\% - 1\%$ is a realistic range. Looking at this error range, all the meshes considered can reach this level with the high-order discretization. However, the best linear solution, which is obtained only on the acute meshes at a very sensitive optimal aspect ratio, can barely compete with the worst $p = 3$ solution. Therefore, higher-order discretizations are generally more efficient at achieving required error levels than their lower-order counterparts.

![Figure 11. Normalized drag error versus DOF for the Navier-Stokes equations](image)
VI. Conditioning of the Linear System

One important concern of the large-angle elements is their numerical conditioning, which dictates the efficiency of the iterative linear solvers applied. This section compares the conditioning of the linear systems obtained from the acute and obtuse meshes, and studies the efficiency of the incomplete LU preconditioning for these meshes. The Jacobian matrix defined in (5) will still be denoted by \( K \), and \( M \) will denote the mass matrix defined by \( M_{ij} = \int_\Omega \phi_i^T \phi_j \, dx \), where \( \{ \phi_i \} \) is the basis for the function space \( V^p_H \) defined in Section II.

VI.A. Eigenvalue Spectrum

All eigenvalue calculations in this section are for the advection-diffusion problem. Table 1 compares the ratios of the maximum and minimum (by moduli) eigenvalues of \( M^{-1}K \) from the acute and obtuse meshes. For all the aspect ratios considered, the acute and obtuse meshes have similar conditioning.

<table>
<thead>
<tr>
<th>Mesh</th>
<th>( \mathcal{A} = 10^3 ), 512 elements</th>
<th>( \mathcal{A} = 10^4 ), 500 elements</th>
<th>( \mathcal{A} = 10^5 ), 800 elements</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p )</td>
<td>( \hat{\kappa} )</td>
<td>( \hat{\kappa} )</td>
<td>( \hat{\kappa} )</td>
</tr>
<tr>
<td>1</td>
<td>12.6</td>
<td>8.3</td>
<td>1.47</td>
</tr>
<tr>
<td>3</td>
<td>97.2</td>
<td>75.4</td>
<td>1.41</td>
</tr>
<tr>
<td>1</td>
<td>263.1</td>
<td>386.2</td>
<td>1.47</td>
</tr>
<tr>
<td>3</td>
<td>1241.3</td>
<td>1746.0</td>
<td>1.41</td>
</tr>
<tr>
<td>1</td>
<td>10365.1</td>
<td>16590.9</td>
<td>1.60</td>
</tr>
<tr>
<td>3</td>
<td>48448.4</td>
<td>60505.5</td>
<td>1.25</td>
</tr>
</tbody>
</table>

Table 1. Eigenvalues of \( M^{-1}K \)

Figure 12 shows the spectra for meshes \( \mathcal{A} = 10^4 \). The spectrum of \( M^{-1}K \) is a measure of the ellipticity of the underlying problem and is independent of the type of interpolation basis chosen. Clearly, the problem is diffusion dominated as the eigenvalues are mainly real. In addition, the spectra for different aspect ratios indicate that the contribution of the diffusion term becomes more important as the aspect ratio increases, i.e., as the element \( y \)-spacing decreases for about the same number of elements. Details can be found in Sun.\(^{28}\)

![Figure 12](image)

Figure 12. Spectra of \( M^{-1}K \) using linear interpolation

Another way to compare the conditioning is to look at the eigenvalue distributions of the block-Jacobi preconditioned system, \( D^{-1}K \), where \( D \) is the diagonal blocks of \( K \). The spectrum is expected to fall in the unit circle centered at \((-1, 0)\), and as high-frequency errors can be removed fairly easily through preconditioning, the smallest eigenvalue (by moduli) implies the difficulty in solving the system. Table 2
lists the smallest eigenvalues for different meshes, and it confirms that the acute and obtuse elements have similar conditioning.

<table>
<thead>
<tr>
<th>Mesh</th>
<th>$\lambda$</th>
<th>$\hat{\kappa}$</th>
<th>$\hat{\kappa}$</th>
<th>$\hat{\kappa}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathcal{A} = 10^3$, 512 elements</td>
<td>1 0.21 0.24</td>
<td>8.5 7.3</td>
<td>0 86</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3 0.072 0.096</td>
<td>26.9 19.9</td>
<td>0.74</td>
<td></td>
</tr>
<tr>
<td>$\mathcal{A} = 10^4$, 500 elements</td>
<td>1 0.016 0.010</td>
<td>121.1 193.7</td>
<td>1.60</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3 0.0076 0.0059</td>
<td>262.4 338.9</td>
<td>1.29</td>
<td></td>
</tr>
<tr>
<td>$\mathcal{A} = 10^5$, 800 elements</td>
<td>1 0.00042 0.00032</td>
<td>4714.9 6237.1</td>
<td>1.32</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3 0.00021 0.00024</td>
<td>9294.1 8179.0</td>
<td>0.88</td>
<td></td>
</tr>
</tbody>
</table>

VI.B. Preconditioning Strategies

Although the conditioning of the linear system arising from the two meshes is similar, comparing preconditioned systems has a more practical value as Krylov subspace methods must be preconditioned to perform well. This section studies the performance of ILU(0) and the more general ILUT($p$, $\tau$) on both the acute and the obtuse meshes for the linear systems defined in (5). Because the previous section has concluded that high-order discretization is preferred, only the preconditioning for the $p = 3$ discretization will be studied. For all the results in this section, the linear solver is iterated until the preconditioned residual reaches $10^{-14}$ relative to the initial residual.

VI.B.1. Performance of ILU(0)

A sufficient condition for ILU(0) being the same as an exact factorization is that the linear system being solved is tridiagonal. For the Jacobian matrix arising from the presented DG discretization, this condition is equivalent to the following:

**Condition VI.1.** Each element on the mesh has no more than two neighbors with non-zero coupling.

This condition is equivalent to being able to partition the mesh into independent lines, where within each line, each element has only two neighbors. In practice, this condition can be rarely met. However, ILU(0) can work very efficiently with an appropriate reordering if a mesh is close to satisfying Condition VI.1, i.e., some elements might have a third neighbor with weak coupling instead of zero coupling.

In general, ILU(0) works poorly for diffusion-dominated problems due to the isotropic nature of diffusion, which usually leads to strong couplings between all neighboring elements. However, for highly anisotropic meshes and/or convection-dominated problems, coupling in one direction can be much weaker than in another. Thus, with an appropriate ordering of unknowns, ILU(0) works efficiently on the acute meshes for the model problem studied. Table 3 lists the number of GMRES iterations using ILU(0) for acute and obtuse meshes of different aspect ratios. As shown in the table, ILU(0) works poorly for obtuse meshes, and as aspect ratio increases (i.e. as the contribution of diffusion becomes more important), the efficiency of ILU(0) drops quickly on these meshes. This can be explained by the strength of coupling between neighboring elements, which governs the performance of ILU(0).

<table>
<thead>
<tr>
<th>$\mathcal{A}$</th>
<th>Acute</th>
<th>Obtuse</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^3$, 512 elements</td>
<td>16 29</td>
<td>6 50</td>
</tr>
<tr>
<td>$10^4$, 500 elements</td>
<td>4 183</td>
<td>4 183</td>
</tr>
</tbody>
</table>

Let the coupling weight for an element $\kappa$ with its neighbor $\kappa'$ be the corresponding entry in $K$ normalized by the diagonal entry for $\kappa$, that is, $\eta_{\kappa\kappa'} \equiv K_{\kappa\kappa'}/\kappa_{\kappa\kappa}$, where $K_{ij}$ denotes the $ij$ block in the matrix $K$ measured in a certain norm, for example, the Frobenius norm.
For a diffusion-dominated problem, the coupling weight $\eta_{\kappa \kappa'}$ scales with $(l_f/d_\kappa)^2$, where $l_f$ is the length of the face between $\kappa$ and $\kappa'$, and $d_\kappa$ is a measure of the diameter of the element $\kappa$. A highly anisotropic right triangle has one edge much shorter than the other two, and so for the model problem considered, the coupling weight on this short edge is negligible. Thus, the acute meshes are close to satisfying Condition VI.1. Furthermore, as the aspect ratio increases, the short edge becomes even shorter, and the coupling across this edge becomes even more insignificant, making ILU(0) perform even better. This is consistent with the results in Table 3. On the other hand, an obtuse triangle does not have this property as all its three edges have similar lengths. This is illustrated in Figure 13. As the diffusion term becomes more important, the coupling across each edge on the obtuse meshes becomes stronger. Conversely, if less diffusion were introduced such that the problem became advection dominated, ILU(0) would work equally well on the acute and the obtuse meshes.

In summary, for ILU(0), the key factor for efficiency is not the maximum angle involved, but the connectivity weight with neighbors, or the strength of coupling among elements. If all three edges of one element have equal coupling strength, regardless of the largest angle involved, then Condition VI.1 will be far from being satisfied, and ILU(0) will have difficulties working efficiently. In fact, for the same reason, ILU(0) works poorly for a simple diffusion-dominated problem on equilateral triangles, as reported by Persson.

To further demonstrate that angle size by itself is not a determining factor in the efficiency of ILU(0), two diffusion cases with anisotropic diffusivity tensors are presented. Both cases use the meshes of $\mathcal{R} = 10^4$ with 500 elements. An analysis of coupling weights for anisotropic diffusion can be found in Sun.

The first case is a pure diffusion problem with a diffusivity tensor given by

$$\tilde{\mu} = \begin{bmatrix} 1 \times 10^{-8} & 2 \times 10^{-12} \\ 2 \times 10^{-12} & 4 \times 10^{-16} \end{bmatrix}. \tag{9}$$

With this diffusivity, the diffusion is zero across one edge of each element on the obtuse meshes, and the coupling is shown in Figure 14.

Table 4 lists the ratio of maximum and minimum eigenvalues, $\lambda_{\text{max}}/\lambda_{\text{min}}$, for the unpreconditioned system, $M^{-1}K$, and also the number of GMRES iterations for the ILU(0) preconditioned system with the MDF reordering. As the table shows, although the system for the obtuse mesh is conditioned only a little better than that for the acute one, ILU(0) works very efficiently for the obtuse mesh as argued based on the strength of coupling among elements.
Table 4. Performance of ILU(0) for anisotropic diffusion in (9) on meshes of $R = 10^4, 500$ elements

<table>
<thead>
<tr>
<th>$\lambda_{\text{max}}/\lambda_{\text{min}}$ of $M^{-1}K$</th>
<th>GMRES It. with ILU(0)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acute</td>
<td>Obtuse</td>
</tr>
<tr>
<td>$9.44 \times 10^4$</td>
<td>$5.69 \times 10^4$</td>
</tr>
<tr>
<td>81</td>
<td>1</td>
</tr>
</tbody>
</table>

The second case has a diffusivity tensor given by

$$\bar{\mu} = \begin{bmatrix} 10^{-8} & 0 \\ 0 & \mu \end{bmatrix},$$

(10)

where $\mu$ controls the diffusivity in the $y$-direction. As it decreases, the relative importance of the diffusivity in the $x$-direction increases, and so the coupling strength across the short edges on the acute meshes increases.

Table 5 lists $\lambda_{\text{max}}/\lambda_{\text{min}}$ for the unpreconditioned system, and also the number of GMRES iterations for the ILU(0) preconditioned system. As less diffusion is introduced in the $y$-direction, the unpreconditioned system becomes better conditioned. However, ILU(0) works less efficiently as the three edges of each element become equally weighted.

Table 5. Performance of ILU(0) for anisotropic diffusion in (10) on acute mesh of $R = 10^4, 500$ elements

<table>
<thead>
<tr>
<th>$\mu$</th>
<th>$\lambda_{\text{max}}/\lambda_{\text{min}}$ of $M^{-1}K$</th>
<th>GMRES It. with ILU(0)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^{-8}$</td>
<td>$3.12 \times 10^5$</td>
<td>3</td>
</tr>
<tr>
<td>$10^{-12}$</td>
<td>$3.11 \times 10^5$</td>
<td>30</td>
</tr>
<tr>
<td>$10^{-16}$</td>
<td>$2.14 \times 10^4$</td>
<td>74</td>
</tr>
</tbody>
</table>

VI.B.2. Preconditioning on Obtuse Meshes

Knowing that ILU(0) works efficiently for the acute meshes, it is desirable to develop an equally efficient preconditioner for the obtuse meshes. $h$- and/or $p$-multigrid preconditioners are known to work well for diffusion-dominated problems, and they have been applied in a DG context.\cite{13,20,29} However, in this work, the ILUT($p, \tau$) is considered for the obtuse meshes, and its efficiency is compared with the ILU(0) results on the acute and obtuse meshes.

A fill-in level of two is considered in this work and is found to be temporally more efficient than one. A higher level is deemed to be too expensive due to memory concerns. Table 6 lists the ILU(0) and ILUT
results for the acute and obtuse meshes. As shown in the table, ILU(0) works poorly for the obtuse meshes as explained previously. The ILUT preconditioner results in a significant improvement. For the Navier-Stokes equations, the computational cost on the obtuse meshes using ILUT is comparable to that on the acute meshes using ILU(0). Also, as ILU(0) works very efficiently on the acute meshes, especially for the advection-diffusion equation, ILUT does not lead to an obvious improvement on the acute meshes, and thus ILUT results on these meshes are not included.

<table>
<thead>
<tr>
<th>Mesh</th>
<th>Preconditioner</th>
<th>Advection-Diffusion Equation</th>
<th>Navier-Stokes Equations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>GMRES Iter.</td>
<td>Time</td>
</tr>
<tr>
<td>Acute</td>
<td>ILU(0)</td>
<td>6</td>
<td>0.108</td>
</tr>
<tr>
<td>Obtuse</td>
<td>ILU(0)</td>
<td>50</td>
<td>0.324</td>
</tr>
<tr>
<td>Obtuse</td>
<td>ILUT</td>
<td>11</td>
<td>0.173</td>
</tr>
</tbody>
</table>

\( A = 10^4 \), 500 elements. Solution order \( p = 3 \).

Linear system from the converged solution state.

Total wall clock time of the linear solver including reordering, factorization, and GMRES, measured on an Intel Xeon 5130 processor at 2.00GHz. O/S: Red Hat Enterprise Linux 4.

ILUT with a fill-in level of two and a threshold of \( \tau = 10^{-6} \).

Let the wall clock time of the linear solve using ILU(0) on the acute meshes be \( t_a = t_f + t_{\text{GMRES}}^a \), where \( t_f \) denotes the factorization time and \( t_{\text{GMRES}}^a \) is the GMRES time. Similarly, let the wall clock time of the linear solve using ILUT on the obtuse meshes be \( t_o = t_f + t_{\text{GMRES}}^o \). Figure 15 plots the ratio of \( t_o/t_a \) versus different drop thresholds for ILUT with a fill-in level of two. With a value of \( \tau = 10^{-6} \), \( t_o \) is about 60% higher (i.e. slower) than \( t_a \) for the advection-diffusion equation. However, it is about 15% higher for the Navier-Stokes equations, for which more complicated characteristics and element-to-element coupling strengths are involved compared to the advection-diffusion equation and so ILU(0) on the acute meshes does not work as efficiently. To further illustrate, Figure 16 shows the ratio of \( t_f/t_f \) and \( t_{\text{GMRES}}^o/t_{\text{GMRES}}^a \) for the two equations. As seen, the factorization time ratio, \( t_f/t_f \), is significantly higher for the Navier-Stokes equations as bigger matrix block sizes are involved. On the other hand, the GMRES time ratio, \( t_{\text{GMRES}}^o/t_{\text{GMRES}}^a \), is higher for the advection-diffusion equation as ILU(0) is close to an exact factorization for this equation on the acute meshes. Since GMRES time is dominant in the total linear solver time, the difference between \( t_a \) and \( t_o \) is much smaller for the Navier-Stokes equations than for the advection-diffusion equation. Also, note that for each of the model equations, there is a critical value of \( \tau \), above which the GMRES time significantly increases, indicating that important information is dropped in the factorization stage.

VII. Conclusion

In this work, the impact of triangle shapes, including large angles and aspect ratios, on accuracy and stiffness for simulations of highly anisotropic problems has been studied. With a high-order DG discretization, it is seen that the discretization error is not sensitive to the maximum angles involved, but using a correct aspect ratio is critical. In addition, the linear system induced from obtuse meshes has similar conditioning with that from acute meshes. However, the solution schemes for these two meshes differ as they involve different strengths of coupling among elements. For this reason, ILU(0) has been shown less efficient on obtuse meshes for an isotropic diffusion problem. With an ILUT preconditioner, the solution on obtuse meshes can be achieved within a comparable time as on acute meshes. Therefore, for anisotropic meshing with high-order discretizations, while aspect ratio is critical, large angles can be overcome through small increases in preconditioning costs.

Acknowledgments

This work was supported by funding from The Boeing Company with technical monitor Dr. Mori Mani.
Figure 15. Wall clock time for linear solver using ILUT on obtuse meshes normalized by the linear solver time using ILU(0) on acute meshes

Figure 16. Factorization time and GMRES time using ILUT on obtuse meshes normalized by the corresponding time using ILU(0) on acute meshes

References


