Optimal trace inequality constants for interior penalty discontinuous Galerkin discretisations of elliptic operators using arbitrary elements with non-constant Jacobians

A.R. Owens, J. Kópházi, M.D. Eaton

Nuclear Engineering Group, Department of Mechanical Engineering, City and Guilds Building, Imperial College London, Exhibition Road, South Kensington, London, SW7 2AZ, United Kingdom

Abstract

In this paper, a new method to numerically calculate the trace inequality constants, which arise in the calculation of penalty parameters for interior penalty discretisations of elliptic operators, is presented. These constants are provably optimal for the inequality of interest. As their calculation is based on the solution of a generalised eigenvalue problem involving the volumetric and face stiffness matrices, the method is applicable to any element type for which these matrices can be calculated, including standard finite elements and the non-uniform rational B-splines of isogeometric analysis. In particular, the presented method does not require the Jacobian of the element to be constant, and so can be applied to a much wider variety of element shapes than are currently available in the literature. Numerical results are presented for a variety of finite element and isogeometric cases. When the Jacobian is constant, it is demonstrated that the new method produces lower penalty parameters than existing methods in the literature in all cases, which translates directly into savings in the solution time of the resulting linear system. When the Jacobian is not constant, it is shown that the naive application of existing approaches can result in penalty parameters that do not guarantee coercivity of the bilinear form, and by extension, the stability of the solution. The method of manufactured solutions is applied to a model reaction-diffusion equation with a range of parameters, and it is found that using penalty parameters based on the new trace inequality constants result in better conditioned linear systems, which can be solved approximately 11% faster than those produced by the methods from the literature.

Keywords: Trace inverse inequality, Interior penalty, Discontinuous Galerkin, Finite element analysis, Isogeometric analysis, Diffusion synthetic acceleration

1. Introduction

Penalty methods were initially introduced by Nitsche [1] as a method to weakly enforce Dirichlet boundary conditions in variational formulations. Interior penalty (IP) terms for discontinuous Galerkin discretisations were first introduced by Wheeler [2] for second-order elliptic problems in a collocation finite element scheme, and for parabolic systems by Arnold [3]. The motivation behind these early schemes was the increased flexibility available in the mesh compared to continuous, conforming finite element methods [3], with advantages related to domain decomposition methods emerging later [4].

Interior penalty methods have been applied using finite elements to both Navier-Stokes problems [5–7] and reaction-diffusion equations; see, for example, Epshteyn et al. [8], and references therein. More recently, these methods have also been applied to reaction-diffusion equations utilising discontinuous Galerkin Isogeometric Analysis (IGA) [9, 10].

In all of these schemes, the penalty parameter on each face must be chosen to be sufficiently large in order to guarantee the coercivity of the associated bilinear form [2, 3, 6, 8, 11]. However, the spectral condition number of

Email address: a.owens12@imperial.ac.uk (A.R. Owens)

the resulting linear system grows linearly with the penalty parameters [12]. Therefore, taking arbitrarily large values for the penalty parameters is not a viable strategy, as this would significantly increase the linear system solution time when using iterative solvers. For the aforementioned problems, in practice, it is desirable to pick the smallest penalty parameters possible while still guaranteeing coercivity of the bilinear form, in order to minimise the solution time of the resulting linear system.

Of particular interest to the authors is the application of IGA to the Modified Interior Penalty (MIP) scheme for Diffusion Synthetic Acceleration (DSA) of discrete ordinates transport calculations [13]. In this case, the scheme is most "consistent" with the discrete ordinates equations being accelerated if the penalty parameters are taken to be $\frac{1}{4}$, and therefore accelerate these equations more efficiently. However, the penalty parameters must still be chosen to be large enough to guarantee coercivity of the bilinear form.

Regardless of the application, it is therefore desirable to calculate the smallest possible penalty parameters that guarantee coercivity; either to minimise the solution time, or to increase the efficiency of the acceleration scheme when using the MIP scheme for DSA. The calculation of these penalty parameters require trace inequalities that bound integrals of the gradients of functions from the finite element function space over element faces f, by integrals of the same quantity over the entire element e multiplied by a constant $C_{e,f}$. The problem of minimising the penalty parameters then reduces to the problem of minimising the Trace Inequality Constant (TIC) $C_{e,f}$.

TICs have been calculated in the literature for simplices of arbitrary polynomial order and dimension by Warburton and Hesthaven [14], and by Hillewaert for quadrilaterals, hexahedra, wedges and pyramids of arbitrary polynomial order [15]. However, the TICs computed in these references bound the integrals of functions from the finite element space, rather than their gradients. This has two major implications.

The first, is that the TICs, and therefore penalty parameters, computed are only valid if the gradients of functions in the finite element function space are also members of the finite element function space. This is only the case if the mapping from the reference element to the physical element is linear, and therefore the Jacobian is constant. This is always the case for linear simplices, but is not generally the case for any other element types, and is certainly not the case when using the Non-Uniform Rational B-Splines (NURBS) of IGA.

The second, is that while the TICs derived in references [14] and [15] are sharp for the finite element function spaces considered, it is actually the gradients of these functions that are of interest in IP methods. Some information has therefore been discarded in their derivation, and sharper bounds may be obtained by taking this information into account.

In this paper, optimal TICs will be derived which take into account the fact that it is the integrals of gradients of functions that must be bound in order to guarantee coercivity of the bilinear form. In addition, these TICs do not require the mapping from the reference element to the physical element to be linear, and so are suitable for use in isogeometric analysis, as well as with general finite elements. Unlike the methods presented in the literature so far [14–16], the TICs derived here do not have a closed-form solution based on the element type, polynomial order, or in the case of Evans et al. [16], constants related to the mapping from the reference to the physical element. Instead, they are calculated numerically for each face of each element in the mesh by solving a generalised eigenvalue problem, and as such can be applied to any element type.

The remainder of this paper is organised as follows. In Section 2, a model reaction-diffusion partial differential equation (PDE) will be stated and its IP bilinear form derived, with a focus on how the TICs impact the penalty parameter values. This will be followed by a brief review of existing methods for calculating these TICs, along with the presentation of a general framework (that does not require a constant Jacobian) that these methods can be considered to be special cases of. As this framework involves the solution of a generalised eigenvalue problem involving the face and volumetric mass matrices, this will be referred to as the "mass matrix method" in the remainder of the paper.

In Section 3, the new TICs will be derived that take into account the fact that we are interested in the gradients of functions from the finite element function space, and that do not rely on the assumption of a constant Jacobian. Rather than doing this on an element type basis, these TICs will be calculated numerically by solving a generalised eigenvalue problem involving the face and volumetric stiffness matrices, and as such will be referred to as the "stiffness matrix method" for the remainder of the paper. In Section 4, numerical results will be presented comparing TICs calculated by methods from the literature, the mass matrix method and the stiffness matrix method, for a variety of polynomial and NURBS element types, both with a constant and a varying Jacobian, and in two and three spatial dimensions. A manufactured solution will also be used to compare the efficiency of solving the resulting linear system when the

penalty parameters are computed using literature methods and the stiffness matrix method. In addition, a high-level overview of the implementation details of the stiffness matrix method will be given in Appendix A.

2. Model problem and literature review

2.1. Model problem

The model problem considered in this work is a reaction-diffusion PDE with a mixture of prescribed Dirichlet and Neumann boundary conditions:

$$-\nabla \cdot (D(\mathbf{r})\nabla\phi(\mathbf{r})) + \Sigma_a(\mathbf{r})\phi(\mathbf{r}) = Q(\mathbf{r}) \qquad \mathbf{r} \in \Omega \tag{1a}$$

$$\phi(\mathbf{r}) = g_d(\mathbf{r}) \qquad \mathbf{r} \in \partial\Omega_D \tag{1b}$$

$$D\nabla\phi(\mathbf{r}) \cdot \mathbf{n} = g_N(\mathbf{r}) \qquad \mathbf{r} \in \partial\Omega_N \tag{1c}$$

where the domain Ω has boundary $\partial\Omega = \partial\Omega_D \cup \partial\Omega_N$ (bold symbols represent vectors throughout this paper, and matrices are underlined). The domain Ω is then partitioned into elements V_e, such that $\Omega = \bigcup_e V_e$. These elements can be polynomial as in standard FEM, NURBS as in IGA, piecewise linear discontinuous elements [17, 18], or any other element type, including meshes that contain hanging-nodes. For brevity, the function space associated with all of these element types will be referred to as the finite element function space in the remainder of this paper. The diffusion coefficient, macroscopic absorption cross-section and source term are assumed to be piecewise constant over the elements, such that:

$$D(\mathbf{r})|_{V_e} = D^e \tag{2a}$$

$$\Sigma_a(\mathbf{r})|_{V_e} = \Sigma_a^e \tag{2b}$$

$$Q(\mathbf{r})|_{V_e} = Q^e \tag{2c}$$

These assumptions simplify the derivation of the penalty parameters, but do not impact the TIC calculation. In particular, it is noted that a more complex form for the diffusion coefficient can be taken into account using the method proposed by Drosson et al. [6]. Defining S_h to be the finite element function space, Equation 1a is then multiplied by a test function $v(\mathbf{r}) \in S_h$ and integrated over each element V_e. The divergence theorem is then applied to the diffusive term, the result is summed over all of the elements in the mesh, and the jump in the current is eliminated to maintain consistency [19] to obtain:

$$\sum_e \int_{V_e} D^e \nabla\phi \cdot \nabla v + \Sigma_a^e \phi v \, d\mathbf{r} - \sum_{f \in \Gamma_I \cup \partial\Omega_D} \int_f [[v]] \cdot \{\{D\nabla\phi\}\} \, dS = \sum_e \int_{V_e} Q^e v \, d\mathbf{r} + \sum_{f \in \partial\Omega_N} \int_f v \, D\nabla\phi \cdot \mathbf{n} \, dS \tag{3}$$

where f represents a face index, and Γ_I is the union of the internal faces of the mesh. The average and jump operators are defined to be:

$$\{\{u\}\} = \begin{cases} \frac{1}{2}(u^+ + u^-) \text{ for } f \in \Gamma_I \\ u \text{ for } f \in \partial\Omega \end{cases} \tag{4a}$$

$$[[u]] = \begin{cases} u^+\mathbf{n}^+ + u^-\mathbf{n}^- \text{ for } f \in \Gamma_I \\ u\mathbf{n} \text{ for } f \in \partial\Omega \end{cases} \tag{4b}$$

The next step is to incorporate the boundary conditions. For Neumann boundaries, Equation 1c can be used to eliminate the $D\nabla\phi \cdot \mathbf{n}$ term in favour of $g_N(\mathbf{r})$ for $f \in \partial\Omega_N$. Dirichlet boundary conditions are weakly enforced using the bilinear form:

$$\sum_{f \in \partial\Omega_D} \mu_f \int_f \phi v \, dS = \sum_{f \in \partial\Omega_D} \mu_f \int_f g_D v \, dS \tag{5}$$

where $\mu_f > 0$ is the penalty parameter for face f. Adding Equation 5 to Equation 3 and incorporating the Neumann boundary condition as described above, we obtain:

$$\begin{aligned}\sum_e \int_{V_e} D^e \nabla\phi \cdot \nabla v + \Sigma_a^e \phi v \, d\mathbf{r} - \sum_{f\in\Gamma_I\cup\partial\Omega_D} \int_f [[v]] \cdot \{\{D\nabla\phi\}\} \, dS + \sum_{f\in\partial\Omega_D} \mu_f \int_f \phi v \, dS \\ = \sum_e \int_{V_e} Q^e v \, d\mathbf{r} + \sum_{f\in\partial\Omega_N} \int_f v g_N \, dS + \sum_{f\in\Omega_D} \mu_f \int_f g_D v \, dS\end{aligned} \tag{6}$$

The penalisation weak form for internal faces is given by:

$$\sum_{f\in\Gamma_I} \mu_f \int_f [[\phi]] \cdot [[v]] \, dS = 0 \tag{7}$$

which weakly enforces continuity of the scalar flux between elements. Adding this to Equation 6 gives the Incomplete Interior Penalty (IIP) weak form. The Symmetric Interior Penalty (SIP) and Nonsymmetric Interior Penalty (NIP) weak forms are obtained by subtracting and adding the weak forms given by:

$$\sum_{f\in\Gamma_I} \int_f [[\phi]] \cdot \{\{D\nabla v\}\} \, dS = 0 \tag{8a}$$

$$\sum_{f\in\partial\Omega_D} \int_f D^e \phi \nabla v \cdot \mathbf{n} \, dS = \sum_{f\in\partial\Omega_D} \int_f D^e g_D \nabla v \cdot \mathbf{n} \, dS \tag{8b}$$

respectively. Note that the incorporation of the weak forms defined in Equations 5, 7 and 8 are consistent with the original Equation 1, as $\phi = g_D$ on $\partial\Omega_D$ and $[[\phi]] = 0$ inside Ω. All three of these variants of the IP method can be written in a single equation by introducing the parameter θ:

$$\begin{aligned}\sum_e \int_{V_e} D^e \nabla\phi \cdot \nabla v + \Sigma_a^e \phi v \, d\mathbf{r} - \sum_{f\in\Gamma_I\cup\partial\Omega_D} \int_f [[v]] \cdot \{\{D\nabla\phi\}\} + \theta[[\phi]] \cdot \{\{D\nabla v\}\} \, dS + \sum_{f\in\Gamma_I\cup\partial\Omega_D} \mu_f \int_f [[\phi]] \cdot [[v]] \, dS \\ = \sum_e \int_{V_e} Q^e v \, d\mathbf{r} + \sum_{f\in\partial\Omega_N} \int_f v g_N \, dS + \sum_{f\in\Omega_D} \mu_f \int_f g_D v \, dS - \theta \sum_{f\in\Omega_D} \int_f D^e g_D \nabla v \cdot \mathbf{n} \, dS\end{aligned} \tag{9}$$

where θ is equal to -1, 0 or 1 for the NIP, IIP and SIP schemes respectively. The left-hand side of Equation 9 defines the bilinear form $a(\phi, v)$ and the right-hand side defines the linear form $l(v)$. The SIP bilinear form is symmetric, while the NIP and IIP bilinear forms are unsymmetric. The bilinear form is coercive provided:

$$a(u,u) \geq c_s \|u\|_h^2 \quad \forall u \in S_h \tag{10}$$

where $\|.\|_h$ is a mesh-dependent norm and c_s is a constant independent of the mesh. If the bilinear form is not coercive, the resulting system can be unstable [11] and optimal convergence rates may not be achieved [8]. $\|.\|_h$ is taken to be the energy norm associated with the bilinear form a:

$$\|u\|_h^2 = \sum_e \int_{V_e} D^e (\nabla u)^2 + \Sigma_a^e u^2 \, d\mathbf{r} + \sum_{f\in\Gamma_I\cup\partial\Omega_D} \mu_f \int_f [[u]]^2 \, dS \tag{11}$$

where a vector quantity squared is taken to mean the dot product of that vector with itself. In the NIP scheme, $a(u,u) = \|u\|_h^2$, and so the inequality in Equation 10 is satisifed for any penalty parameters $\mu_f > 0$ with $c_s \leq 1$. However, optimal error estimates in the L_2 norm are yet to be proven for this scheme [8], and no trace inequalities are required, and so the NIP scheme will not be discussed further in this work. In the IIP and SIP schemes, the left-hand side of Equation 10 is given by:

$$a(u,u) = \sum_e \int_{V_e} D^e (\nabla u)^2 + \Sigma_a^e u^2 \, d\mathbf{r} - (1+\theta) \sum_{f\in\Gamma_I\cup\partial\Omega_D} \int_f [[u]] \cdot \{\{D\nabla u\}\} \, dS + \sum_{f\in\Gamma_I\cup\partial\Omega_D} \int_f \mu_f [[u]]^2 \, dS \tag{12}$$

The first and last terms are "contributing" to the coercivity as they are always positive, and so it is the possibly negative second term that must be bound from below in order to guarantee coercivity overall. The penalty parameters on each face μ_f provide the required degrees of freedom to satisfy the inequality in Equation 10, and are calculated as follows. Define

$$\mathcal{N}(u) := \sum_e \int_{V_e} D^e(\nabla u)^2 + \Sigma_a^e u^2 \, d\mathbf{r} \tag{13}$$

as these terms remain unchanged throughout many steps of the penalty parameter derivation. Following the methodology of Shahbazi [11]:

$$a(u,u) \geq \mathcal{N}(u) - (1+\theta) \sum_{f \in \Gamma_I \cup \partial\Omega_D} \frac{1}{2\epsilon_f} \int_f \{\{D\nabla u\}\}^2 \, dS + \sum_{f \in \Gamma_I \cup \partial\Omega_D} \left(\mu_f - \frac{(1+\theta)\epsilon_f}{2}\right) \int_f [[u]]^2 \, dS \tag{14a}$$

$$= \mathcal{N}(u) - (1+\theta) \sum_{f \in \Gamma_I} \frac{1}{8\epsilon_f} \int_f \left[(D^+\nabla u^+)^2 + (D^-\nabla u^-)^2 + 2(D^+\nabla u^+) \cdot (D^-\nabla u^-) \right] dS - ... \tag{14b}$$

$$... - (1+\theta) \sum_{f \in \partial\Omega_D} \frac{1}{2\epsilon_f} \int_f (D\nabla u)^2 \, dS + \sum_{f \in \Gamma_I \cup \partial\Omega_D} \left(\mu_f - \frac{(1+\theta)\epsilon_f}{2}\right) \int_f [[u]]^2 \, dS$$

$$\geq \mathcal{N}(u) - (1+\theta) \sum_{f \in \Gamma_I} \frac{1}{4\epsilon_f} \int_f \left[(D^+\nabla u^+)^2 + (D^-\nabla u^-)^2 \right] dS - (1+\theta) \sum_{f \in \partial\Omega_D} \frac{1}{2\epsilon_f} \int_f (D\nabla u)^2 \, dS + ... \tag{14c}$$

$$... + \sum_{f \in \Gamma_I \cup \partial\Omega_D} \left(\mu_f - \frac{(1+\theta)\epsilon_f}{2}\right) \int_f [[u]]^2 \, dS$$

$$= \sum_e \left[\int_{V_e} D^e(\nabla u)^2 + \Sigma_a^e u^2 \, d\mathbf{r} - (1+\theta) \left(\sum_{f \in \partial V_e \cap \Gamma_I} \frac{(D^e)^2}{4\epsilon_f} \int_f (\nabla u)^2 \, dS + \sum_{f \in \partial V_e \cap \partial\Omega_D} \frac{(D^e)^2}{2\epsilon_f} \int_f (\nabla u)^2 \, dS \right) \right] + ...$$

$$... + \sum_{f \in \Gamma_I \cup \partial\Omega_D} \left(\mu_f - \frac{(1+\theta)\epsilon_f}{2}\right) \int_f [[u]]^2 \, dS \tag{14d}$$

The first line in Equation 14 follows from Equation 12 by using Young's inequality in the form:

$$\mathbf{a} \cdot \mathbf{b} \leq \frac{\epsilon \mathbf{a}^2}{2} + \frac{\mathbf{b}^2}{2\epsilon} \quad \forall \epsilon > 0 \tag{15}$$

with $\mathbf{a} = [[u]]$ and $\mathbf{b} = \{\{D\nabla u\}\}$. The second line is obtained by expanding $\{\{D\nabla u\}\}$ using the definition of the face average operator, and the third line follows by another application of Young's inequality with $\epsilon = 1$. The fourth line is equivalent to the third, and is obtained by representing the integrals over internal and Dirichlet faces as integrals over element boundaries, as well as the fact that the diffusion coefficient is constant within each element.

The next step in the derivation requires a trace inequality in the form:

$$\int_f (\nabla u)^2 \, dS \leq C_{e,f} \int_{V_e} (\nabla u)^2 \, d\mathbf{r} \quad \forall u \in S_h \tag{16}$$

where f is a face of V_e and $C_{e,f}$ is the trace inequality constant. Existing methods for calculating the $C_{e,f}$ will be discussed in Section 2.2, while the calculation of the sharpest possible values is the subject of this paper, and will be covered in Section 3. To continue with the penalty parameter derivation, however, it is sufficient to assume that the

constant $C_{e,f}$ exists. Equation 14 then continues:

$$a(u,u) \geq \sum_e \left[\int_{V_e} D^e (\nabla u)^2 + \Sigma_a^e u^2 \, d\mathbf{r} - (1+\theta) \left(\sum_{f \in \partial V_e \cap \Gamma_I} \frac{(D^e)^2 C_{e,f}}{4\epsilon_f} \int_{V_e} (\nabla u)^2 \, d\mathbf{r} + \sum_{f \in \partial V_e \cap \partial\Omega_D} \frac{(D^e)^2 C_{e,f}}{2\epsilon_f} \int_{V_e} (\nabla u)^2 \, d\mathbf{r} \right) \right] + ...$$
$$... + \sum_{f \in \Gamma_I \cup \partial\Omega_D} \left(\mu_f - \frac{(1+\theta)\epsilon_f}{2} \right) \int_f [[u]]^2 \, dS \tag{17a}$$

$$= \sum_e \left[\left(1 - (1+\theta) \left(\sum_{f \in \partial V_e \cap \Gamma_I} \frac{D^e C_{e,f}}{4\epsilon_f} + \sum_{f \in \partial V_e \cap \partial\Omega_D} \frac{D^e C_{e,f}}{2\epsilon_f} \right) \right) \int_{V_e} D^e (\nabla u)^2 \, d\mathbf{r} + \int_{V_e} \Sigma_a^e u^2 \, d\mathbf{r} \right] + ... \tag{17b}$$
$$... + \sum_{f \in \Gamma_I \cup \partial\Omega_D} \left(\mu_f - \frac{(1+\theta)\epsilon_f}{2} \right) \int_f [[u]]^2 \, dS$$

Coercivity therefore requires:

$$1 - (1+\theta) \left(\sum_{f \in \partial V_e \cap \Gamma_I} \frac{D^e C_{e,f}}{4\epsilon_f} + \sum_{f \in \partial V_e \cap \partial\Omega_D} \frac{D^e C_{e,f}}{2\epsilon_f} \right) > 0 \quad \text{and} \tag{18a}$$

$$\mu_f - \frac{(1+\theta)\epsilon_f}{2} > 0 \tag{18b}$$

$$\implies \mu_f > \frac{(1+\theta)\epsilon_f}{2} \tag{18c}$$

As θ, D^e and the $C_{e,f}$ are known constants, the (positive) ϵ_f must be chosen on each face so that Equation 18a is satisifed for every element. The penalty parameters μ_f then trivially follow from Equation 18c, and so it is clear that values of ϵ_f that are as small as possible are desirable, in order to minimise the corresponding penalty parameters. The complexity arises as the ϵ_f are defined for each *face*, which in general belong to more than one element, while Equation 18a must be satisfied for every *element*. In general, this is a very challenging optimisation problem, and it is not immediately clear what a "good" solution should look like.

However, a sufficient condition is to use an "anisotropic" length scale [6], in which each term in the sum in Equation 18a is bound separately, such that Equation 18a is guaranteed to hold. Defining N_e to be the number of faces f of element V_e that are not on a Neumann boundary:

$$1 - (1+\theta) \left(\sum_{f \in \partial V_e \cap \Gamma_I} \frac{D^e C_{e,f}}{4\epsilon_f} + \sum_{f \in \partial V_e \cap \partial\Omega_D} \frac{D^e C_{e,f}}{2\epsilon_f} \right) > 0 \tag{19a}$$

$$\iff \frac{1}{1+\theta} > \sum_{f \in \partial V_e \cap \Gamma_I} \frac{D^e C_{e,f}}{4\epsilon_f} + \sum_{f \in \partial V_e \cap \partial\Omega_D} \frac{D^e C_{e,f}}{2\epsilon_f} \tag{19b}$$

$$\impliedby \frac{1}{N_e(1+\theta)} > \begin{cases} \dfrac{D^e C_{e,f}}{4\epsilon_f} & \forall f \in \partial V_e \cap \Gamma_I \\ \dfrac{D^e C_{e,f}}{2\epsilon_f} & \forall f \in \partial V_e \cap \partial\Omega_D \end{cases} \tag{19c}$$

$$\iff \epsilon_f > \begin{cases} (1+\theta)\dfrac{D^e C_{e,f} N_e}{4} & \forall e \ni f \notin \partial\Omega_D \\ (1+\theta)\dfrac{D^e C_{e,f} N_e}{2} & \forall f \in \partial\Omega_D \end{cases} \tag{19d}$$

As this must hold for every element that contains face f, the maximum is taken over all elements that contain face f, and the final penalty parameters for internal faces are given by:

$$\mu_f > \frac{(1+\theta)^2}{8} \max_{e \ni f} \{ D^e C_{e,f} N_e \} \tag{20}$$

The expression is simpler on Dirichlet boundaries, as each face belongs to only one element, and so:

$$\mu_f > \frac{(1+\theta)^2}{4} D^e C_{e,f} N_e \tag{21}$$

These penalty parameters are proportional to the TICs $C_{e,f}$, and so any saving that can be made by calculating the minimal possible TIC values translate directly into savings in the penalty parameters. It is noted that the IIP penalty parameters are $\times 4$ lower than those of the SIP scheme. It is also noted that if the definition of $\{\{u\}\}$ on faces $f \in \partial\Omega$ in Equation 4 was $\frac{1}{2}u$, rather than u, which is physically reasonable, the factor of two difference between the expressions appearing in Equations 20 and 21 would not be present.

2.2. Literature Review

The penalty parameter derivation in the previous section relied on the trace inequality and associated constant $C_{e,f}$ defined in Equation 16. However, the trace inequalities for simplices, quadrilaterals, hexahedra, wedges and pyramids in the literature appear in the form [14, 15]:

$$\int_f u^2 \, dS \leq C_{e,f} \int_{V_e} u^2 \, d\mathbf{r} \quad \forall u \in S_h \tag{22a}$$

$$C_{e,f} := \tilde{C}_{e,f} \frac{\mathcal{A}(f)}{\mathcal{V}(V_e)} \tag{22b}$$

where $\mathcal{A}(f)$ is the area of face f and $\mathcal{V}(V_e)$ is the volume of element V_e. Equation 22a holding implies that Equation 16 holds, under the additional assumption that $\nabla u \in S_h$. This is only the case in general if the mapping from the reference element to the physical element is linear, or equivalently, if the Jacobian matrix is constant across the element. This is guaranteed for linear simplices, but is not generally the case for other element types or polynomial orders. For simplices, quadrilaterals and hexahedra, the values of $\tilde{C}_{e,f}$ are given by:

$$\tilde{C}_{e,f} = \begin{cases} \dfrac{(p+1)(p+d)}{d} & \text{for simplices in } d \text{ dimensions} \\ (p+1)^2 & \text{for quadrilaterals and hexahedra} \end{cases} \tag{23}$$

The TICs for simplices have been employed by Shahbazi [11] and by Epshteyn et al. [8] (with a slight improvement taking into account the element shapes) in the context of SIP schemes for reaction-diffusion equations. It was found that the penalty parameters computed using the TICs from Equation 23 are $\approx \times 3$ larger than those required for stability in the solution [11]. This suggests that explicitly taking into account the fact that it is the the integrals of gradients that are required in Equation 16 may be beneficial even when the Jacobian is constant, and this will be demonstrated numerically in Section 4.

However, in the general case when the Jacobian matrix is not constant:

1. Equation 22a offers no information about the inequality of interest for IP schemes defined in Equation 16.
2. The TICs in Equation 23 are no longer valid.

For three dimensional, tensor-product NURBS with no internal knots (of which hexahedral finite elements are a special case), Evans et al. provide explicit bounding constants of the form in Equation 22a [16]. These do not require a constant Jacobian, and instead take the mapping from the reference element to the physical element into account through various infinity norms of the Jacobian matrix and its inverse. The drawbacks of this method are:

1. The results presented in reference [16] only apply to three dimensional NURBS and hexahedral finite elements. As with the constant Jacobian results in Equation 23, similar results must be derived separately for each different element type under consideration.
2. The computed bounds are not sharp.

Comparisons in reference [16] are made to sharp bounds that are computed via a Rayleigh quotient approach [20]. This is the general framework that will be outlined in the following subsection, of which the methods used to compute the explicit TICs in Equation 23 are special cases. In addition, the stiffness matrix method presented in Section 3 follows the same principles as the mass matrix method, but with added complications, and so it is useful to see the simpler mass matrix method first.

2.3. Mass matrix method

Expanding the function $u(\mathbf{r})$ in terms of the N finite element basis functions for an element:

$$u(\mathbf{r}) = \sum_{i=1}^{N} u_i R_i(\mathbf{r}) \tag{24}$$

and defining the column vectors $\mathbf{u}$ and $\mathbf{R}(\mathbf{r})$ in the natural manner such that $u(\mathbf{r}) = \mathbf{u}^T\mathbf{R} = \mathbf{R}^T\mathbf{u}$, then the function u^2 can be written as:

$$u^2 = \mathbf{u}^T\mathbf{R}\mathbf{R}^T\mathbf{u} \tag{25}$$

As $\mathbf{u}$ is a constant vector, Equation 22a can be expressed as:

$$\mathbf{u}^T\left(\int_f \mathbf{R}\mathbf{R}^T\, dS\right)\mathbf{u} \leq C_{e,f}\,\mathbf{u}^T\left(\int_{V_e} \mathbf{R}\mathbf{R}^T\, d\mathbf{r}\right)\mathbf{u} \tag{26}$$

Defining the volumetric and face mass matrices as:

$$\underline{M}_v = \int_{V_e} \mathbf{R}\mathbf{R}^T\, d\mathbf{r} \tag{27a}$$

$$\underline{M}_f = \int_f \mathbf{R}\mathbf{R}^T\, dS \tag{27b}$$

respectively, Equation 26 can be written as:

$$\mathbf{u}^T\underline{M}_f\mathbf{u} \leq C_{e,f}\,\mathbf{u}^T\underline{M}_v\mathbf{u} \tag{28}$$

$\underline{M}_v$ is the standard volumetric mass matrix, and as such is Symmetric Positive-Definite (SPD), while the face mass matrix $\underline{M}_f$ is Symmetric Positive SemiDefinite (SPSD). Therefore the generalised eigenvalue problem:

$$\left(\underline{M}_f - \lambda\underline{M}_v\right)\mathbf{x} = \mathbf{0} \quad \text{with normalisation} \tag{29a}$$

$$\mathbf{x}^T\underline{M}_v\mathbf{x} = 1 \tag{29b}$$

has N distinct eigenvalues that are all ≥ 0, and their associated eigenvectors are linearly independent [21]. Defining $\underline{X}$ to be the matrix whose columns are the right eigenvectors of Equation 29 and $\underline{\Lambda} = \text{diag}(\lambda_1, ..., \lambda_N)$, $\underline{M}_f$ and $\underline{M}_v$ can be simultaneously diagonalised:

$$\underline{X}^T\underline{M}_f\,\underline{X} = \underline{X}^T\underline{M}_v\,\underline{X}\,\underline{\Lambda} \quad \text{where} \tag{30a}$$

$$\underline{X}^T\underline{M}_v\,\underline{X} = \underline{I}_N = \underline{X}\,\underline{M}_v\,\underline{X}^T \tag{30b}$$

$$\implies \underline{M}_v = \underline{X}^{-1}(\underline{X}^T)^{-1} \quad \text{and} \tag{30c}$$

$$\underline{X}^T\underline{M}_f\,\underline{X} = \underline{\Lambda} = \underline{X}\,\underline{M}_f\,\underline{X}^T \tag{30d}$$

$$\implies \underline{M}_f = \underline{X}^{-1}\underline{\Lambda}(\underline{X}^T)^{-1} \tag{30e}$$

where $\underline{I}_N$ is the identity matrix of size N. The second equalities in the second and fourth lines follow from the symmetry of $\underline{M}_v$ and $\underline{M}_f$ respectively. Note that $\underline{X}^T \neq \underline{X}^{-1}$ in general, due to the normalisation condition in Equation 29. The right eigenvectors $\underline{X}$ form a basis of $\mathbb{R}^N$, and so by changing basis such that $\mathbf{u} = \underline{X}^T\mathbf{v}$, the trace inequality constant in Equation 22a can be calculated as:

$$\int_f u^2\, dS = \mathbf{u}^T\underline{M}_f\mathbf{u} \tag{31a}$$

$$= \mathbf{u}^T\underline{X}^{-1}\underline{\Lambda}(\underline{X}^T)^{-1}\mathbf{u} \tag{31b}$$

$$= \mathbf{v}^T\underline{\Lambda}\mathbf{v} \tag{31c}$$

$$= \sum_{i=1}^{N} \lambda_i v_i^2 \tag{31d}$$

$$\leq \max_i(\lambda_i) \sum_{i=1}^{N} v_i^2 \tag{31e}$$

$$= \max_i(\lambda_i)\, \mathbf{v}^T\mathbf{v} \tag{31f}$$

$$= \max_i(\lambda_i)\, \mathbf{u}^T \underline{X}^{-1}(\underline{X}^T)^{-1}\mathbf{u} \tag{31g}$$

$$= \max_i(\lambda_i)\, \mathbf{u}^T \underline{M}\mathbf{u} \tag{31h}$$

$$= \max_i(\lambda_i) \int_{V_e} u^2\, d\mathbf{r} \tag{31i}$$

Therefore, $C_{e,f} = \max_i(\lambda_i)$, the maximum eigenvalue of the generalised eigenvalue problem defined by the volumetric and face mass matrices. It can easily be seen that the bound is sharp, as equality is attained for the eigenvector corresponding to the largest eigenvalue.

The methods derived in the constant Jacobian case for simplices, quadrilaterals, hexahedra, wedges and pyramids rely on the existence of an orthonormal basis of the finite element function space S_h, which exactly corresponds to finding the generalised eigenvectors such that $\underline{X}^T \underline{M}_v \underline{X} = \underline{I}_N$. Properties of the face mass matrix $\underline{M}_f$ are then exploited in order to calculate closed-form expressions for the TICs such as those in Equation 23 [14, 15].

Existing methods in the literature appear to be focused around finding such closed-form expressions. However, in the methods of Warburton et al. [14] and Hillewaert [15], this both limits the application in IP methods to elements with constant Jacobians, and does not take into account the fact that the IP methods require bounds on the integrals of gradients. In the case of the methods presented in Evans et al. [16], the bounds are not sharp.

In the following Section, TICs will be derived that are both provably sharp for the IP application, and do not require a constant Jacobian. These will require the solution of a generalised eigenvalue problem of a similar form to Equation 29. The authors believe that computing the $C_{e,f}$ in this manner is a viable approach for a broad range of applications, and will both increase the flexibility in the mesh by including elements with a non-constant Jacobian, and reduce the penalty parameters in the IP bilinear form to their minimal possible values.

3. Trace inequality constant derivation

The trace inequality of interest is repeated here for clarity:

$$\int_f (\nabla u)^2\, dS \leq C_{e,f} \int_{V_e} (\nabla u)^2\, d\mathbf{r} \quad \forall u \in S_h \tag{32}$$

Calculating the optimal TICs required for the IIP and SIP methods in Equation 32 via the stiffness matrix method begins as in the mass matrix method, by expanding the function $u(\mathbf{r})$ in terms of the finite element basis functions:

$$u(\mathbf{r}) = \sum_{i=1}^{N} u_i R_i(\mathbf{r}) \tag{33a}$$

$$\implies \nabla u(\mathbf{r}) = \sum_{i=1}^{N} u_i \nabla R_i(\mathbf{r}) \tag{33b}$$

The column vectors $\mathbf{u}$ and $\mathbf{R}(\mathbf{r})$, and the tensor $\nabla\mathbf{R}(\mathbf{r})$ are defined in the natural manner, such that $u(\mathbf{r}) = \mathbf{u}^T\mathbf{R} = \mathbf{R}^T\mathbf{u}$, and $\nabla u(\mathbf{r}) = \mathbf{u}^T\nabla\mathbf{R} = \nabla\mathbf{R}^T\mathbf{u}$ Then the function $(\nabla u)^2$ can be defined as:

$$(\nabla u)^2 = \mathbf{u}^T \nabla\mathbf{R}(\nabla\mathbf{R})^T\mathbf{u} \tag{34}$$

As $\mathbf{u}$ is a constant vector, Equation 32 can be expressed as:

$$\mathbf{u}^T \left(\int_f \nabla\mathbf{R}(\nabla\mathbf{R})^T\, dS \right) \mathbf{u} \leq C_{e,f}\, \mathbf{u}^T \left(\int_{V_e} \nabla\mathbf{R}(\nabla\mathbf{R})^T\, d\mathbf{r} \right) \mathbf{u} \tag{35}$$

Defining the volumetric and face stiffness matrices as:

$$\underline{S}_v = \int_{V_e} \nabla \mathbf{R} (\nabla \mathbf{R})^T \, d\mathbf{r} \tag{36a}$$

$$\implies (\underline{S}_v)_{i,j} = \int_{V_e} \nabla R_i \cdot \nabla R_j \, d\mathbf{r} \tag{36b}$$

$$\underline{S}_f = \int_{f} \nabla \mathbf{R} (\nabla \mathbf{R})^T \, dS \tag{36c}$$

$$\implies (\underline{S}_f)_{i,j} = \int_{f} \nabla R_i \cdot \nabla R_j \, dS \tag{36d}$$

respectively, Equation 35 can be written as:

$$\mathbf{u}^T \underline{S}_f \mathbf{u} \leq C_{e,f} \, \mathbf{u}^T \underline{S}_v \mathbf{u} \tag{37}$$

In Section 2.3 the fact that $\underline{M}_v$ is SPD was used to calculate $C_{e,f}$ as the maximum eigenvalue of the generalised eigenvalue problem given in Equation 29. However, $\underline{S}_v$ is only SPD if the constant function is not a member of the finite element space S_h. However, most finite element spaces used in practice require that the constant function *is* contained in S_h, in order for the spatial discretisation to be conservative. In this case, $\underline{S}_v$ is SPSD, with a zero eigenvalue corresponding to the eigenvector $\mathbf{u}_c$ that represents the constant function in S_h. This eigenvalue has multiplicity one, which can be observed from the positivity of the integrand on the right-hand side of Equation 32.

However, any function that is constant over the element is also constant over the face. This complicates matters, as the constant function $\mathbf{u}_c$ is also in the kernel of $\underline{S}_f$ (which is also SPSD), and so the shared kernel of $\underline{S}_v$ and $\underline{S}_f$ is non-trivial. In this case, the pair of matrices is said to be non-regular [21], and the generalised eigenvalue problem:

$$\left(\underline{S}_f - \lambda \underline{S}_v\right) \mathbf{x} = \mathbf{0} \tag{38}$$

does not necessarily have the properties that the regular pair $\underline{M}_v$ and $\underline{M}_f$ did in Section 2.3. This stems from the fact that for any pair of complex numbers (α, β), $|\alpha \underline{S}_f + \beta \underline{S}_v| = 0$ [21], which can be observed by considering the quantity $\mathbf{u}_c^T (\alpha \underline{S}_f + \beta \underline{S}_v) \mathbf{u}_c = \alpha \mathbf{u}_c^T \underline{S}_f \mathbf{u}_c + \beta \mathbf{u}_c^T \underline{S}_v \mathbf{u}_c = 0$.

In this case Saad states: "*This is a special singular problem. In practice, it may sometimes be desirable to remove the singularity, and compute the eigenvalues associated with the restriction of the pair to the complement of the null space. This can be achieved provided we can compute a basis of the common null space, a task that is not an easy one for large sparse matrices, especially if the dimension of the null space is not small.*" [21] While this may be complicated in general, the shared kernel of $\underline{S}_v$ and $\underline{S}_f$ is one-dimensional and is spanned by the vector that represents the constant function $\mathbf{u}_c$.

In Legendre-like, hierarchical bases, the constant function is precisely one of the basis functions of S_h, in which case the column vector $\mathbf{u}_c$ will be equal to zero in all rows but one, labelled k. The shared kernel of $\underline{S}_v$ and $\underline{S}_f$ can then be eliminated simply by deleting row k and column k from the matrices.

Many other finite element bases (including NURBS) form a partition of unity, in which case $\mathbf{u}_c = (1, 1, ..., 1)^T$, although in the general case, $\mathbf{u}_c$ can be easily computed by Galerkin projection. Once $\mathbf{u}_c$ is known, an orthonormal basis of $\mathbb{R}^N$ that contains $\mathbf{u}_c$ as the k^{th} basis vector can be constructed by, for example, modified Gram-Schmidt or Householder transformations. Denote by $\underline{X}$ the $N \times N - 1$ dimensional matrix whose columns are given by these orthonormal basis vectors with the k^{th} column removed. Then $\underline{S}_v$ and $\underline{S}_f$ can then be transformed to the $N - 1$ dimensional space $\mathbb{R}^N \setminus \text{span}\{\mathbf{u}_c\}$ by:

$$\underline{S}'_v = \underline{X}^T \underline{S}_v \underline{X} \tag{39a}$$

$$\underline{S}'_f = \underline{X}^T \underline{S}_f \underline{X} \tag{39b}$$

i.e. the constant function has been orthogonalised out of the finite element space. Denote this reduced order space by S'_h. As the kernel of $\underline{S}_v$ has been removed, $\underline{S}'_v$ is SPD, $\underline{S}'_f$ is SPSD, and all of the analysis for the mass matrix method

described in Section 2.3 holds. Then the trace inequality in this reduced order space:

$$\int_f (\nabla u')^2 \, dS \leq C'_{e,f} \int_{V_e} (\nabla u')^2 \, d\mathbf{r} \quad \forall u' \in S'_h \tag{40}$$

has associated constant $C'_{e,f}$ given by the maximum eigenvalue of the generalised eigenvalue problem:

$$\left(\underline{S}'_f - \lambda \underline{S}'_v\right)\mathbf{x} = \mathbf{0} \quad \text{with normalisation} \tag{41a}$$

$$\mathbf{x}^T \underline{S}'_v \mathbf{x} = 1 \tag{41b}$$

However, every function $u \in S_h$ can be written as $u = u' + c$, where $u' \in S'_h$ and c is a constant. Substituting this sum into Equation 32, and noting that the gradient of a constant is identically zero, implies that $C_{e,f} = C'_{e,f}$. As in the mass matrix method, this TIC is provably sharp, by again choosing the eigenvector corresponding to the largest generalised eigenvalue. However, as the gradient of the function has now been explicitly taken into account, it is expected that this method will give smaller TICs than those in the literature when the Jacobian is constant, as well as being applicable to non-linear mappings, and this will be demonstrated in Section 4.

In practice, for spaces S_h of low dimension N, such as those that are local to a single element, this procedure is not computationally demanding, and can be utilised as presented. All of the numerical results in Section 4 used this method, and the resulting generalised eigenvalue problems were solved with the LAPACK routine DSYGV [22]. However, for applications with large continuous regions, such as multipatch IGA, N can be as large as $O(10^6)$ [10] (or possibly larger). In this case, it would be extremely computationally demanding both to explicitly calculate the orthonormal basis, and to find the largest generalised eigenvalue using dense matrices.

In the mass matrix method, the matrices $\underline{M}_v$ and $\underline{M}_f$ can be computed directly using standard finite element machinery, and stored in a sparse format, such as Compressed Row Storage (CRS). As $\underline{M}_v$ is SPD, computing the largest generalised eigenvalue can be achieved via a standard power iteration, or one of its variants, and conjugate gradient techniques can be utilised, so that the only operations required are sparse matrix-vector multiplications and dot products.

In the stiffness matrix method, the matrices $\underline{S}_v$ and $\underline{S}_f$ can also be computed with standard finite element machinery (possibly with some small modifications). However, $\underline{S}'_v$ and $\underline{S}'_f$ are both computationally expensive to compute, and, in general, dense, precluding their use in a standard power iteration involving only sparse operations. However, it is possible that a power iteration could be employed directly using $\underline{S}_v$ and $\underline{S}_f$, by using a projected conjugate gradient technique for the solution step, in the spirit of the projected Krylov methods presented in the literature [23–25]. In these methods, a power iteration system of the form:

$$\underline{S}_v \mathbf{x}^{l+1} = \underline{S}_f \mathbf{x}^l \tag{42}$$

where l is an iteration index, and $\underline{S}_v$ is singular, can be solved by "projecting out" the known kernel of $\underline{S}_v$, without explicitly forming a basis for the complement of the kernel. As these projection operations are represented by matrix multiplication, methods of this form may be more suitable for use with the stiffness matrix method applied to large sparse matrices.

4. Numerical results

The majority of the results presented in this section compare the trace inequality constants, for a variety of element types and polynomial orders, computed in three different ways: those from the literature in Equation 23, the mass matrix method described in Section 2.3, and the stiffness matrix method presented in Section 3. The elements considered will be split into those that have a constant Jacobian, i.e. a linear mapping from the reference element to the physical element, and those that do not. The TIC calculation methods from the literature are only strictly valid for elements with a constant Jacobian. In this case, it will be demonstrated that TICs calculated by the mass matrix method agree with those from the literature, and also that those calculated using the stiffness matrix method are always the smallest of the three values.

All three methods will also be compared for elements with a non-constant Jacobian. Although the TICs from the literature are not strictly valid in this case, it has been claimed that these values will be close to the true values provided

the elements are not too deformed [15]. In addition, in applications of the MIP scheme, TICs of a similar form to those in Equation 22b are used, where the ratio $\mathcal{A}(f)/\mathcal{V}(V_e)$ is replaced by an "orthogonal length" $1/h_\perp$ [13, 26]. It will be demonstrated that in some cases, the application of these TICs to IP schemes using elements with a non-constant Jacobian will produce large overestimates (compared to the stiffness matrix method) in the penalty parameters required for coercivity, thereby increasing the condition number of the resulting system. Even more seriously, it will also be demonstrated that in some cases, the penalty parameters computed by these methods are lower than those computed by the stiffness matrix method, in which case the resulting linear system is not even guaranteed to be coercive.

The "raw" TICs $C_{e,f}$ from the inequalities given in Equations 22a and 32 presented in the following results have been normalised by multiplying by a factor of $\mathcal{V}(V_e)/\mathcal{A}(f)$. This is to facilitate comparison with the TICs from the literature in Equation 23, which are constant for a given element type and polynomial order after this normalisation.

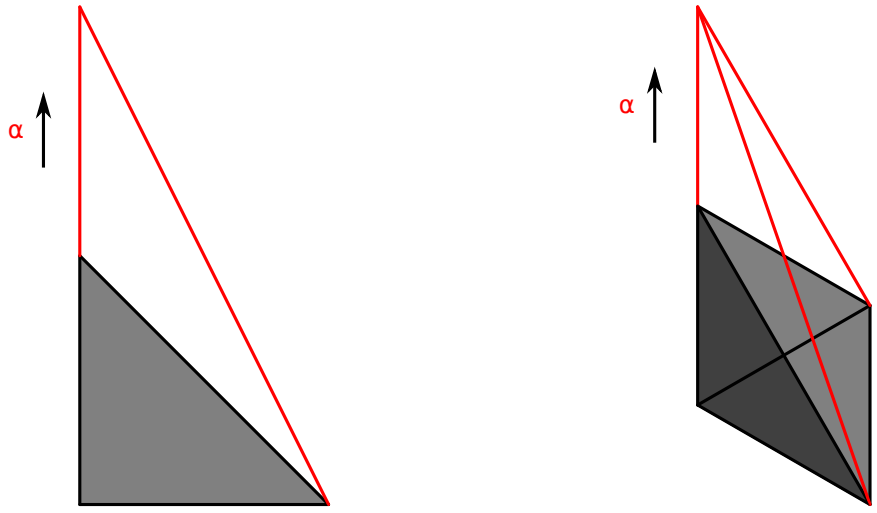
In addition, the Method of Manufactured Solutions (MMS) will be used to compare the linear system solution efficiency when the penalty parameters are computed using TICs from the literature, compared to those computed using the stiffness matrix method. This will involve convergence studies using a variety of material constants in Equation 1a, as well as a selection of meshes of varying regularity.

4.1. Constant Jacobian

4.1.1. Simplices

To create a sequence of triangular elements with a constant Jacobian, an initial triangle is formed whose mapping from the reference element is the identity, i.e. the corner nodes have (x, y) coordinates $(0, 0)$, $(1, 0)$ and $(0, 1)$. The third node is then translated up the y-axis by the vector $(0, \alpha)$, as demonstrated in Figure 1a.

A similar procedure is followed with tetrahedra, where the initial corner nodes have (x, y, z) coordinates $(0, 0, 0)$, $(1, 0, 0)$, $(0, 1, 0)$ and $(0, 0, 1)$. The fourth node is then translated up the z-axis by the vector $(0, 0, \alpha)$, as demonstrated in Figure 1b. For quadratic simplices, the edge nodes lie halfway between the corresponding corner nodes, in order to maintain a linear mapping from the reference element.



(a) Triangle (b) Tetrahedron

Figure 1: The method used to distort triangles (a) and tetrahedra (b) with a constant Jacobian, using a distortion parameter α.

The parameter α was varied from 0 to 1000 for both triangles and tetrahedra. As the Jacobian remains constant, the normalised TICs calculated using the literature method and the mass matrix method remain constant, and this will be true for all of the element types considered with a constant Jacobian. In the case of simplices, this is also true of the normalised TICs calculated by the stiffness matrix method (this is not true of non-simplicial elements even with a constant Jacobian). The results of these calculations are presented in Table 1.

	Triangles		Tetrahedra	
Calculation Method	Linear	Quadratic	Linear	Quadratic
Literature	3	6	$2.\dot{6}$	5
Mass Matrix	3	6	$2.\dot{6}$	5
Stiffness Matrix	1	3	1	$2.\dot{6}$

Table 1: The normalised TICs for linear and quadratic simplex elements. For simplices, these values do not depend on the distortion parameter α when calculated using any of the methods.

As expected, the TICs calculated by the mass matrix method agree with those in the literature in all cases. It can also be seen that TICs calculated using the stiffness matrix method correspond to those from the literature of one polynomial order lower, e.g. the stiffness matrix method TIC for quadratic triangles is the same as the literature TIC for linear triangles. This is also expected, due to the following observation.

Denote by $S^d_{h,p}$ the finite element space of functions defined over a simplex of dimension d and of polynomial order p. Then, provided the mapping from the reference element to the physical element is linear, $u \in S^d_{h,p} \implies \nabla u \in S^d_{h,p-1}$. This identity also holds for $\mathbb{P}$-type finite elements defined over hypercubes, and so the same results should hold, although TICs for these elements will not be explicitly calculated in this paper.

4.1.2. Quadrilaterals

To create a sequence of quadrilateral tensor-product, or $\mathbb{Q}$-type, finite elements with a constant Jacobian (which are necessarily parallelograms) an initial square is formed whose corner nodes have (x, y) coordinates $(0, 0)$, $(1, 0)$, $(0, 1)$ and $(1, 1)$. The third and fourth nodes are then translated in the x-direction by the vector $(\alpha, 0)$, as demonstrated in Figure 2. For quadratic elements, the edge nodes lie halfway between the corresponding corner nodes, in order to maintain a linear mapping from the reference element.

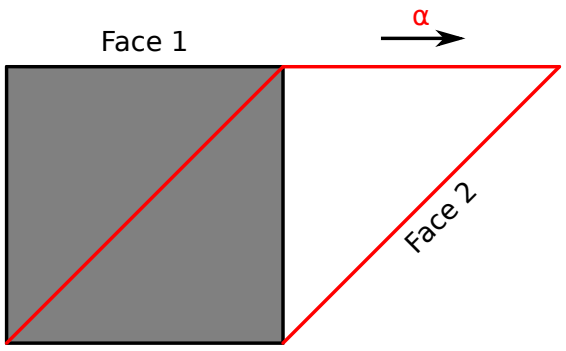
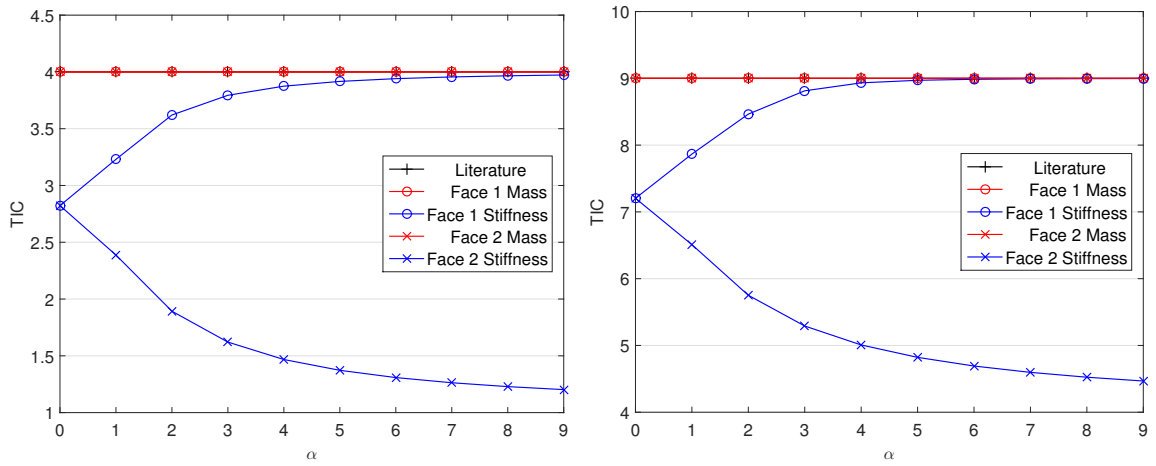


Figure 2: The method used to distort quadrilaterals with a constant Jacobian, using a distortion parameter α. The resulting elements are parallelograms.

The parameter α was varied from 0 to 9, and the results for linear and quadratic elements are presented in Figure 3. The TICs have only been presented for two of the faces, as symmetry dictates that they will be the same for pairs of opposite faces. Face 1 is the top face, and face 2 is one of the slanted edges.

As expected, the TICs calculated by the mass matrix method agree with those in the literature in all cases. For $\alpha = 0$, the physical element is a square, and so the TICs calculated by the stiffness matrix method on face 1 and face 2 are identical. This is not the case, however, as the distortion parameter is increased, at which point the TIC for face 1 increases monotonically, while the TIC for face 2 decreases monotonically.

The TICs for both faces appear to be asymptotically approaching fixed values as α increases, and so to test this the $\alpha = 1000$ case was also computed. The face 1 TICs approach the values computed using the methods from the literature for large α. The face 2 TICs approach 1 and 4 for linear and quadratic elements respectively. Although not



(a) Linears (b) Quadratics

Figure 3: The variation of the TICs for a constant Jacobian quadrilateral (a parallelogram) with linear (a) and quadratic (b) basis functions. The parameter α is a measure of the distortion of the element, demonstrated graphically in Figure 2. The TICs calculated by the literature and mass matrix methods agree for both faces and all α values.

presented here, these values correspond to TICs for line elements of order 1 and 2 respectively, computed using the stiffness matrix method.

In all cases the TICs computed by the stiffness matrix method are lower than those presented in the literature, by up to a factor of 4 for linear elements and 2.25 for quadratic elements. It is also noted that these same factors quantify the difference between the TICs computed for different faces using the stiffness matrix method, while the method from the literature treats all faces identically.

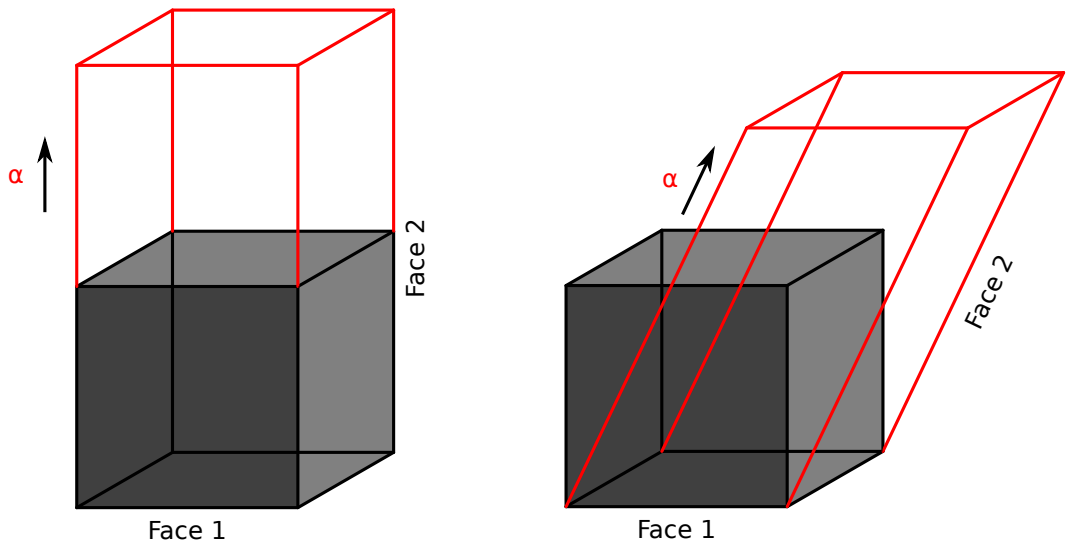
4.1.3. Hexahedra

To create sequences of tensor-product hexahedral elements with a constant Jacobian (which are necessarily parallelepipeds), an initial cube is formed whose corner nodes have (x, y, z) coordinates $(0, 0, 0)$, $(1, 0, 0)$, $(0, 1, 0)$, $(1, 1, 0)$, $(0, 0, 1)$, $(1, 0, 1)$, $(0, 1, 1)$ and $(1, 1, 1)$. Two different methods were then used to distort this cube. In the first, the nodes whose initial z-coordinates are equal to 1 are translated in the z-direction by the vector $(0, 0, \alpha)$, as demonstrated in Figure 4a, creating a square pipe. In the second, the nodes whose initial z-coordinates are equal to 1 are translated by the vector (α, α, α), as demonstrated in Figure 4b. In both cases, the parameter α was varied from 0 to 9. For quadratic hexahedra, the edge nodes lie halfway between the corresponding corner nodes, in order to maintain a linear mapping from the reference element.

The results for the first distortion method are presented in Figure 5 for both linear and quadratic elements. The TICs have only been presented for two of the faces, as symmetry dictates that they will be the same for the two square faces, as well as for the four rectangular faces. Face 1 is the bottom face, which does not change as the distortion parameter α is varied, and face 2 is one of the rectangular faces.

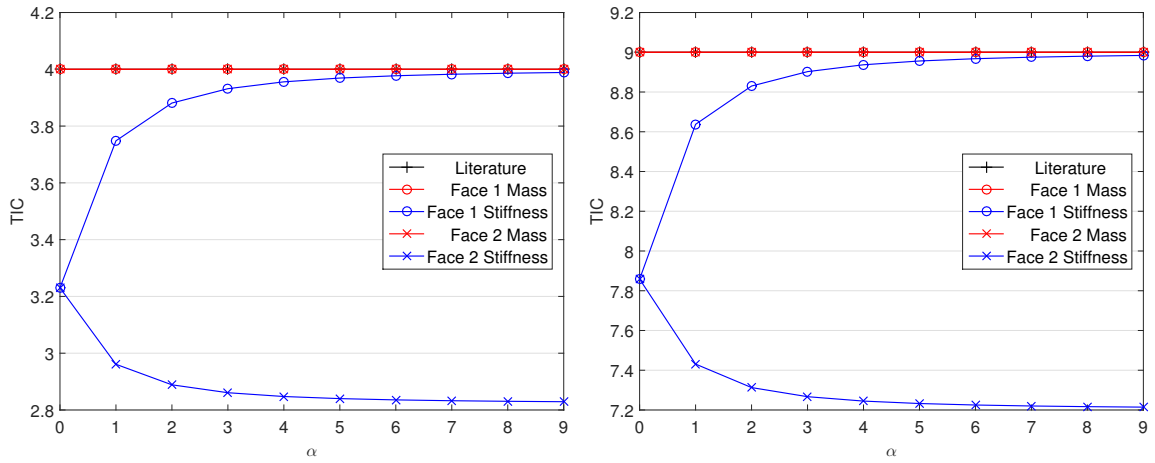
As expected, the TICs calculated by the mass matrix method agree with those in the literature in all cases. For $\alpha = 0$, the physical element is a cube, and so the TICs calculated by the stiffness matrix method on face 1 and face 2 are identical. As α increases, the behaviour is then qualitatively similar to that of the quadrilaterals presented in Section 4.1.2. The face 1 TICs monotonically increase, and asymptotically approach the values computed using the method from the literature. The face 2 TICs monotonically decrease, approaching $2\sqrt{2}$ and 7.2039580694 for linear and quadratic elements respectively. These correspond to the TICs computed by the stiffness matrix method for non-distorted quadrilateral elements of order 1 and 2 respectively.

In all cases the TICs computed by the stiffness matrix method are lower than those presented in the literature, by up to a factor of $\sqrt{2}$ for linear elements and ~ 1.25 for quadratic elements. As with the quadrilaterals in Section 4.1.2, these same factors quantify the difference between the TICs computed for different faces using the stiffness matrix method, while the method from the literature treats all faces identically.



(a) (b)

Figure 4: The two methods used to distort hexahedra with a constant Jacobian, using a distortion parameter α. The resulting elements are parallelepipeds.



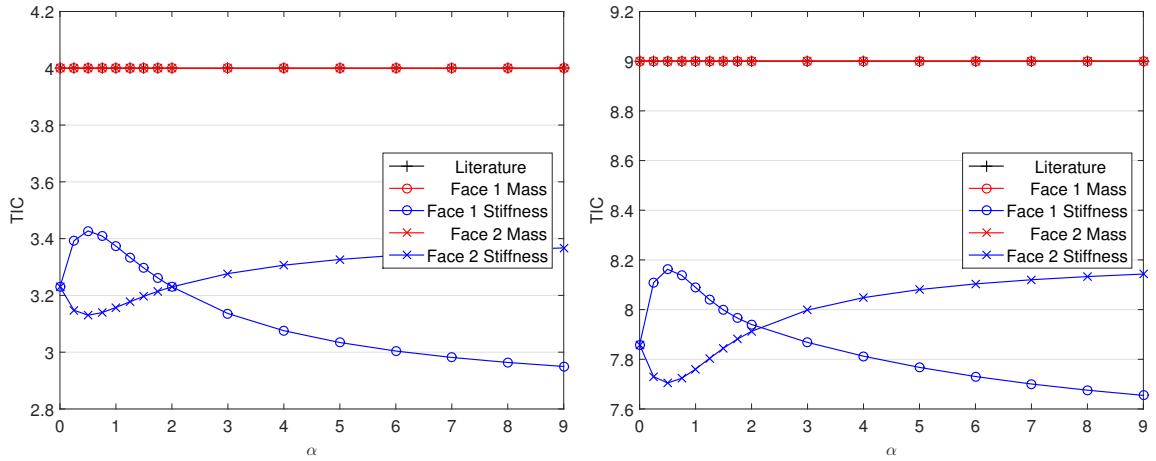
(a) Linears (b) Quadratics

Figure 5: The variation of the TICs for a constant Jacobian hexahedron (a parallelepiped) with linear (a) and quadratic (b) basis functions, distorted by the first method. The parameter α is a measure of the distortion of the element, demonstrated graphically in Figure 4a. The TICs calculated by the literature and mass matrix methods agree for both faces and all α values.

The results for the second distortion method are presented in Figure 6, for both linear and quadratic elements. As with the earlier examples, only the two distinct face types are presented, with face 1 corresponding to the bottom face, which does not change as the distortion parameter α is varied, and face 2 is one of the parallelogram faces.

As expected, the TICs calculated by the mass matrix method agree with those in the literature in all cases. For $\alpha = 0$, the physical element is a cube, and so the TICs calculated by the stiffness matrix method on face 1 and face 2 are identical. In this case, however, the stiffness matrix TICs do not approach those computed by the method from the literature as α increases for either of the faces.

In all cases the TICs computed by the stiffness matrix method are lower than those presented in the literature. For



(a) Linears (b) Quadratics

Figure 6: The variation of the TICs for a constant Jacobian hexahedron (a parallelepiped) with linear (a) and quadratic (b) basis functions, distorted by the second method. The parameter α is a measure of the distortion of the element, demonstrated graphically in Figure 4b. The TICs calculated by the literature and mass matrix methods agree for both faces and all α values.

linear elements, they are a factor of ~ 1.17 to ~ 1.36 lower, while for quadratic elements, they are a factor of ~ 1.10 to ~ 1.18 lower than those calculated by the method from the literature.

4.2. Non-constant Jacobian

4.2.1. Quadratic triangle

To create a sequence of quadratic triangular elements with a non-constant Jacobian, an initial, equilateral triangle was formed, whose corner nodes have (x, y) coordinates $(-\frac{1}{2}, 0)$, $(\frac{1}{2}, 0)$ and $(0, \frac{\sqrt{3}}{2})$. The node at the centre of the bottom face, with initial coordinates $(0, 0)$, is then translated in the negative y-direction by the vector $(0, -\alpha)$, as demonstrated in Figure 7a. The parameter α was varied between 0 and 1.

The results for the two unique faces are presented in Figure 7b, with face 1 representing one of the straight edges that does not change as the parameter α is varied, and face 2 is the edge that is curved when $\alpha > 0$. Note that as the Jacobian is no longer constant, the TICs calculated using the method from the literature and those calculated by the mass matrix method are no longer the same for $\alpha > 0$.

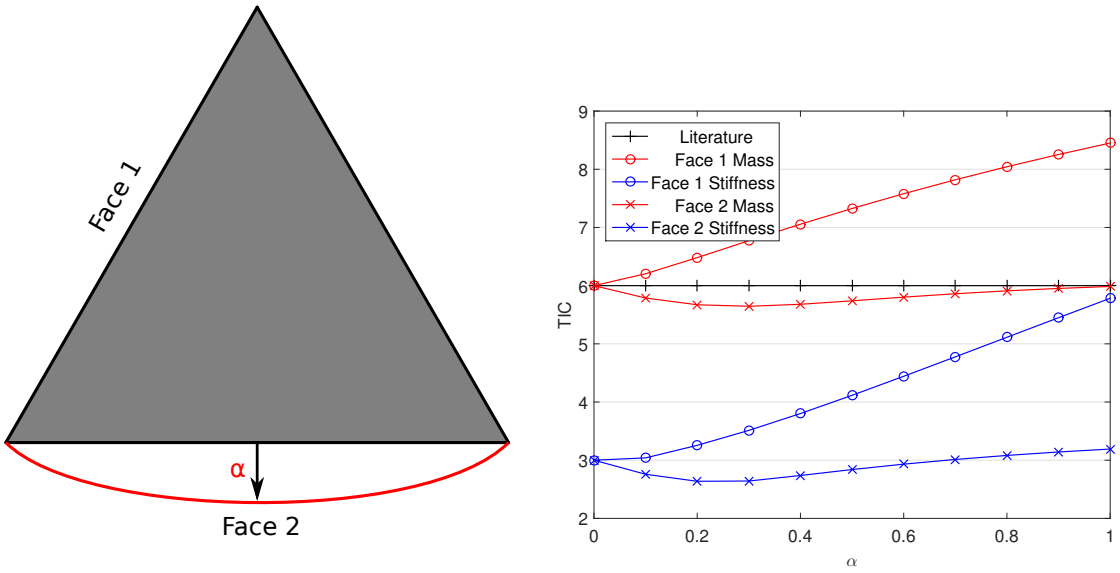
For face 2, the TICs calculated by the mass matrix method and those calculated by the method from the literature are quite similar, while those calculated by the stiffness matrix method are $\sim \times 2$ lower. For face 1, the TICs calculated by the literature method underestimate the true value of $C_{e,f}$ required to bound the face integral in Equation 22a. For the values of α considered, the stiffness matrix method consistently produces lower TICs than those from the literature, and so the penalty parameters produced by this method would be conservative. However, the trends suggest that this would not be the case for values of $\alpha > 1$, in which case the associated bilinear form may no longer be coercive.

4.2.2. Linear quadrilateral

To create a sequence of bilinear quadrilateral elements with a non-constant Jacobian, an initial square was formed, whose corner nodes have (x, y) coordinates $(-1, -1)$, $(1, -1)$, $(-1, 1)$ and $(1, 1)$. The fourth node is then translated by the vector (α, α), as demonstrated in Figure 8a. The parameter α was varied between 0 and 9.

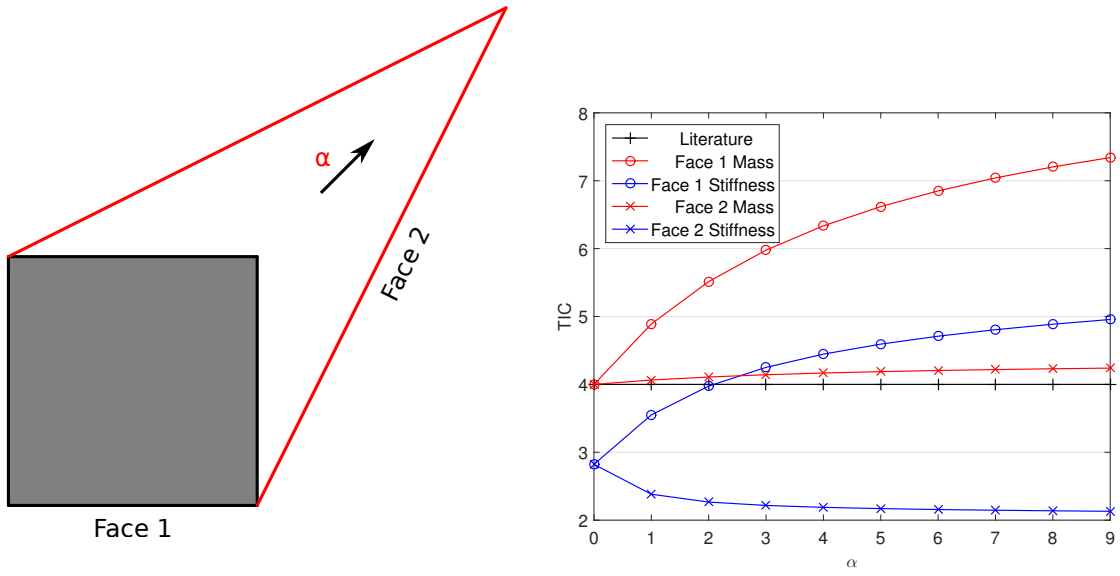
The results for the two unique faces are presented in Figure 8b, with face 1 representing one of the edges that does not change as the parameter α is varied, and face 2 is one of the edges that is stretched as α increases.

For face 2, the TICs calculated by the mass matrix method and those calculated by the method from the literature are quite similar, while those calculated by the stiffness matrix method are $\sim \times 2$ lower. For face 1, the TICs calculated by the literature method underestimate the true value of $C_{e,f}$ required to bound the face integral in Equation 22a. For $\alpha \geq 2$, the TICs calculated by the stiffness matrix method are larger than those calculated by the method from



(a) (b)

Figure 7: The distortion method (a) and TICs (b) for quadratic triangular elements with a non-constant Jacobian.



(a) (b)

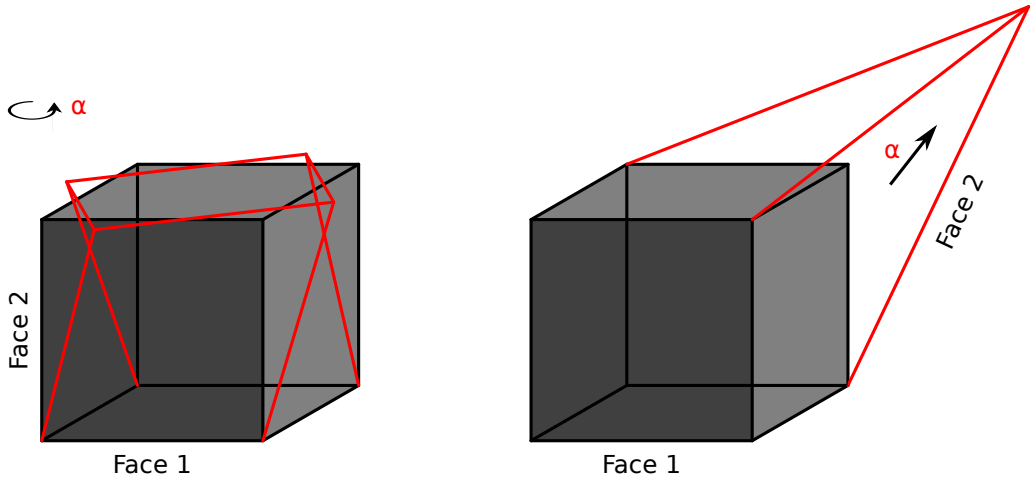
Figure 8: The distortion method (a) and TICs (b) for linear quadrilateral elements with a non-constant Jacobian.

the literature. In this case, penalty parameters based on the method from the literature would not guarantee that the bilinear form is coercive.

4.2.3. Linear Hexahedra

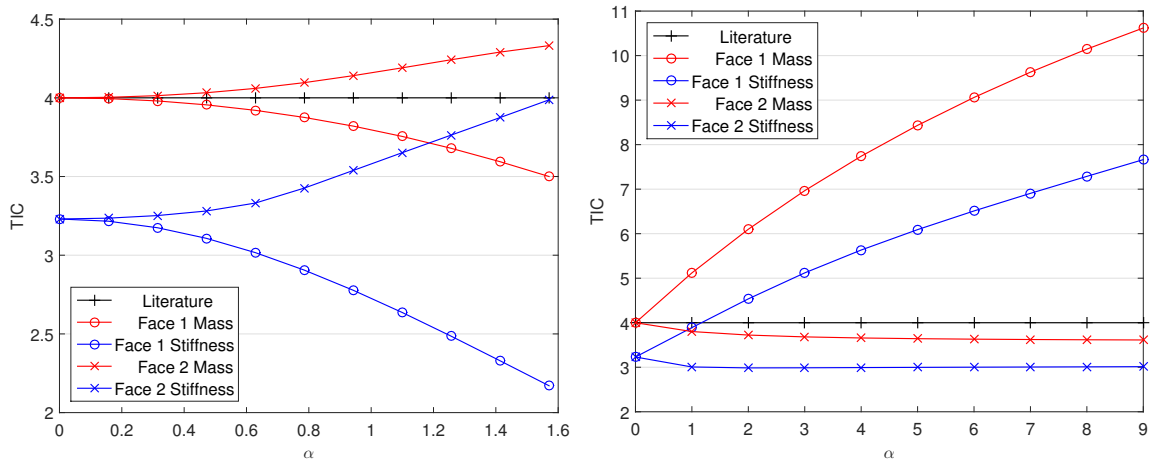
To create sequences of trilinear hexahedral elements with a non-constant Jacobian, an initial cube is formed whose corner nodes have (x, y, z) coordinates $(-1, -1, -1)$, $(1, -1, -1)$, $(-1, 1, -1)$, $(1, 1, -1)$, $(-1, -1, 1)$, $(1, -1, 1)$, $(-1, 1, 1)$

and $(1, 1, 1)$. Two different methods were then used to distort this cube. In the first, the nodes whose initial z-coordinates are equal to 1 are rotated about the z-axis by an angle α, as demonstrated in Figure 9a, with α varying between 0 and $\frac{\pi}{2}$. In the second, the node with initial coordinates $(1, 1, 1)$ is translated by the vector (α, α, α), as demonstrated in Figure 9b, with α varying between 0 and 9.



(a) (b)

Figure 9: The two methods used to distort hexahedra with a non-constant Jacobian, using a distortion parameter α. In the first method (a), the top of a cube is rotated by an angle α around the z-axis. In the second method (b), one of the nodes is translated along a vector directly away from the centre of the cube.



(a) (b)

Figure 10: The variation of the TICs for non-constant Jacobian hexahedra, distorted by the first (a) and second (b) method by a parameter α, as demonstrated in Figure 9.

The results for the first distortion method are presented in Figure 10a for the two unique faces. Face 1 is the face in the $z = -1$ plane, that does not alter as the parameter α is varied, while face 2 is one of the distorted parallelogram faces. For face 1, the literature method slightly overestimates the value of $C_{e,f}$ required to bound the face integral in Equation 22a, while those calculated by the stiffness matrix method are a factor of 1.24 to 1.84 lower than those from the literature. For face 2, the literature method slightly underestimates $C_{e,f}$ compared to the mass matrix method. The

stiffness matrix method for this face produces lower TICs, and therefore penalty parameters, than the method from the literature for the values of α considered here. However, the trend suggests that this would not be the case for more twisted hexahedra, in which case the penalty parameters calculated using the literature methods will not guaranteee coercivity of the bilinear form.

The results for the second distortion method are presented in Figure 10b for the two unique faces. Face 1 is the face in the $z = -1$ plane, that does not alter as the parameter α is varied, while face 2 is one of the distorted faces that has the same shape as those in Figure 8a. For face 2, the mass matrix method and the method from the literature produce quite similar TICs, while the stiffness matrix produces TICs that are $\sim \times 1.33$ lower than the literature method. However, this is not the case at all for face 1. In this case, the TICs calculated by the literature method underestimate those calculated by the mass matrix method by a large margin, so that the constant $C_{e,f}$ in Equation 22a would be vastly underestimated. More seriously, for $\alpha > 1$, the TICs calculated by the stiffness matrix method are larger than those calculated by the literature method, in which case the penalty parameters produced by the literature method would not guarantee coercivity of the bilinear form.

4.2.4. NURBS pincell

This test problem involves the application of IGA to a quarter of a pincell, which frequently arises in reactor physics applications. This geometry is constructed from five biquadratic NURBS patches (elements), as shown in Figure 11. The fuel pin, represented by a quarter circle, is constructed from a single patch. The cladding, represented by a quarter annulus, is constructed from two patches, as is the surrounding moderator. As such, there are three distinct element types in the geometry, as symmetry dictates that the two cladding elements will give the same TICs as one another, as will the two moderator elements. As the TICs for an element do not depend on neighbouring elements (unlike the penalty parameters, see Equation 20), these three element types can be considered separately.

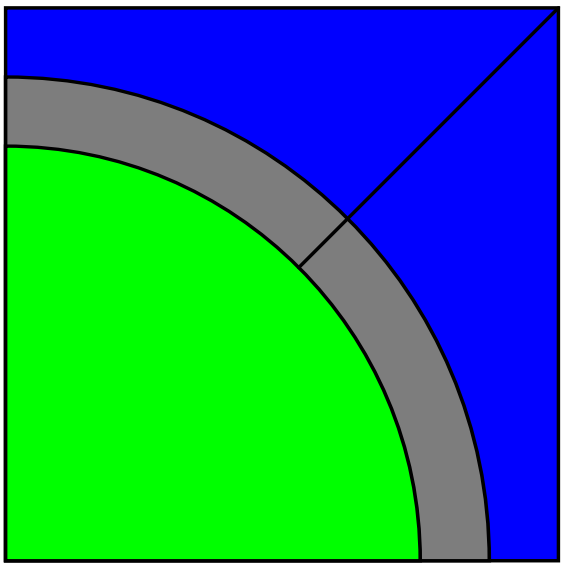


Figure 11: The geometry for a quarter of a pincell, constructed using five biquadratic NURBS patches, which are delineated by the black curves.

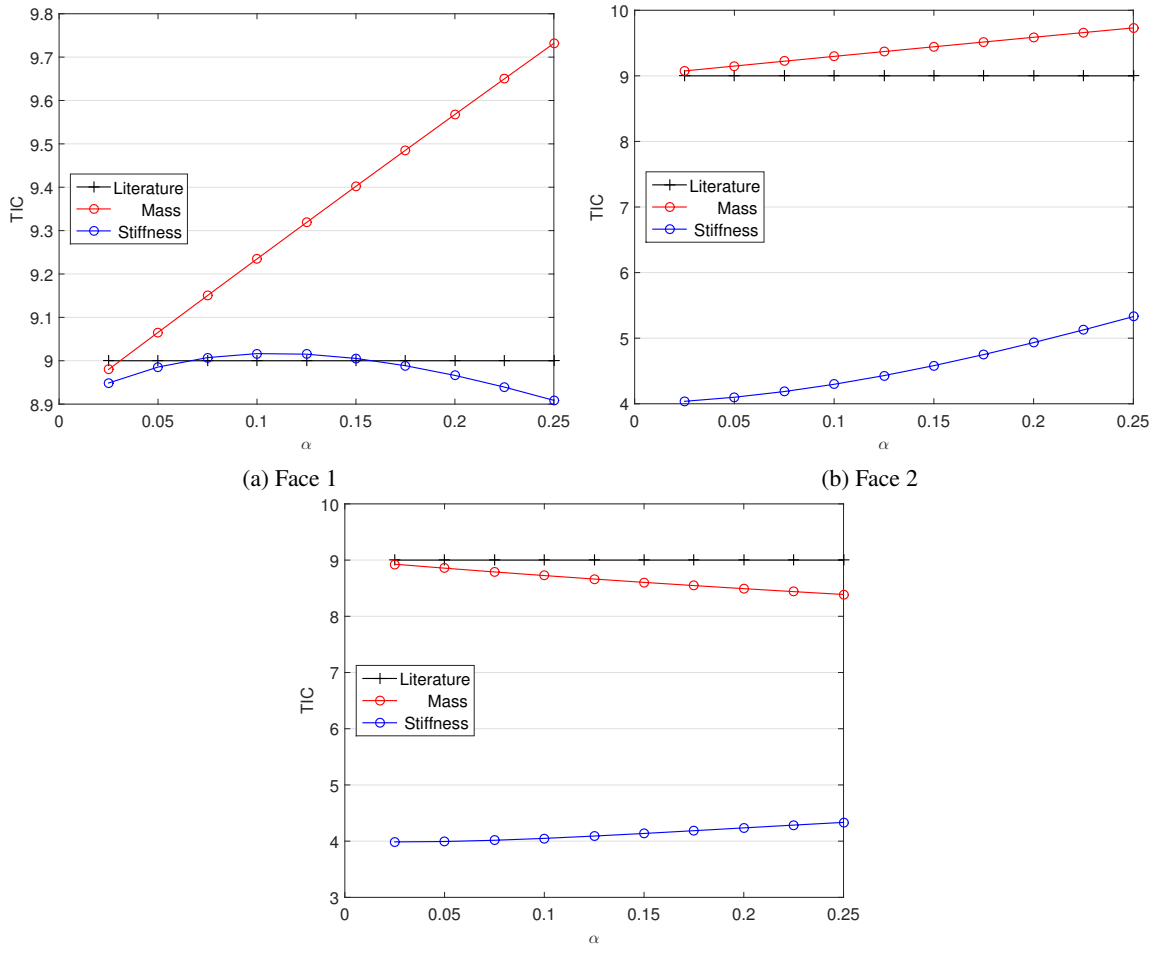
For completeness, TICs will be computed by the literature method from Equation 23, by the mass matrix method and by the stiffness matrix method, although the assumption that the elements are not too deformed [15] is clearly not satisifed in this case. It is noted that the TICs calculated by Evans et al. [16], which will not be considered here, are a non-optimal method for estimating those of the mass matrix method, and so would be larger than those computed by the mass matrix method in all cases.

For the quarter circle, the normalised TICs do not vary as the radius is altered when calculated by any of the three methods, and these constants are presented in Table 2. Face 1 is a straight edge, and face 2 is a circle arc. For face

Calculation Method	Face 1	Face 2
Literature	9	9
Mass Matrix	9.26	18.45
Stiffness Matrix	6.03	445.44

Table 2: The normalised TICs for the two unique faces of a quarter circle, constructed using a biquadratic NURBS patch with no internal knots. For this element, these values do not depend on the radius when calculated using any of the methods.

1, all three calculation methods produce TICs of the same order of magnitude. However, this is not the case at all for the circle arc, in which the mass matrix method and literature method underestimate the TICs by factors of 24 and 50 respectively. In this case, it is clear that any penalty parameters computed based on the inequality in Equation 22a for NURBS are completely inadequate, and the full inequality involving the gradient in Equation 32 must be taken into account to guarantee coercivity of the bilinear form.



(a) Face 1

(b) Face 2

(c) Face 3

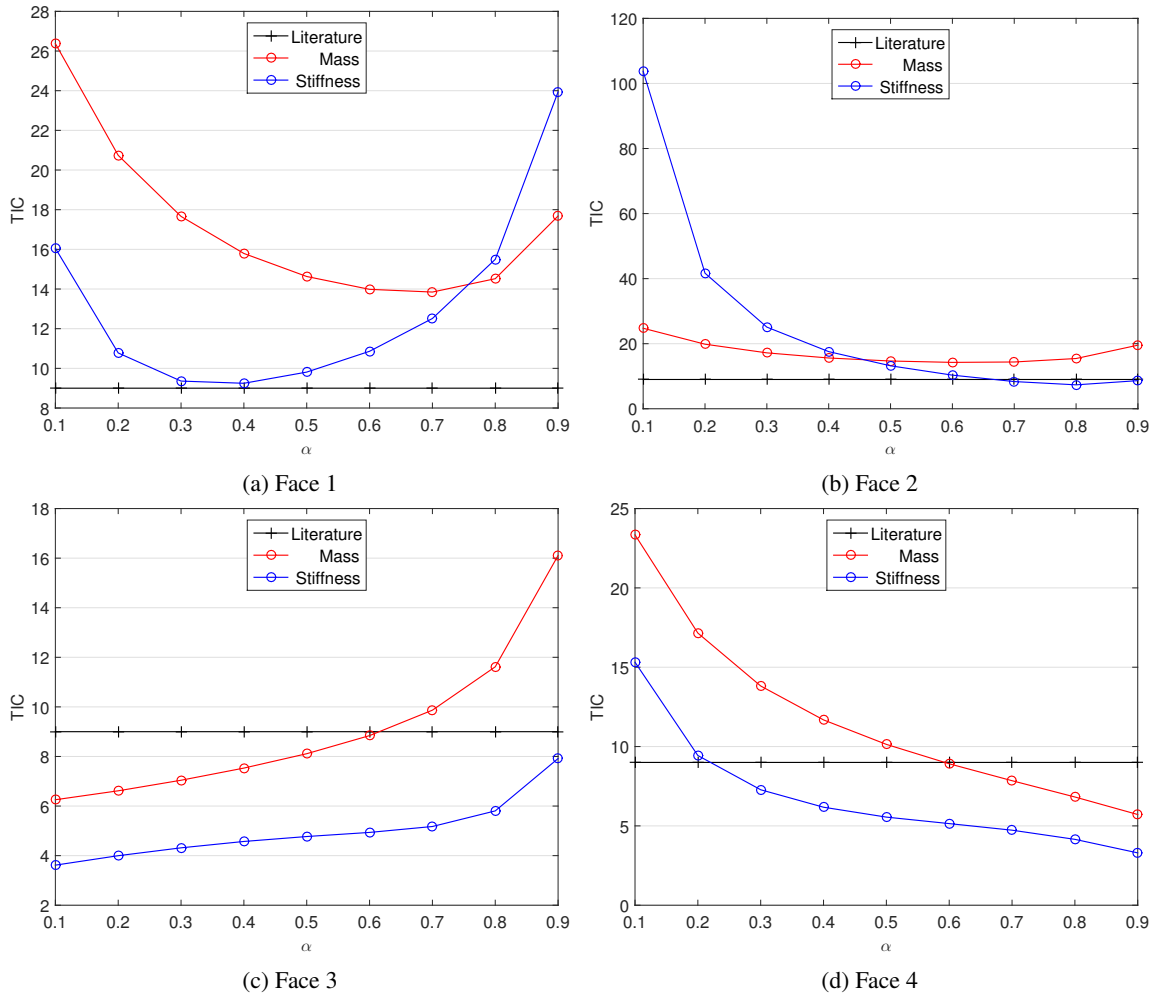
Figure 12: The TICs for the three unique faces of an eighth of an annulus, as the ratio of the outer radius to the inner radius is varied. Face 1 is a straight edge, face 2 is the inner circle arc and face 3 is the outer circle arc.

For the eighth of an annulus element, there are three unique faces, as the straight edges will have the same TICs by symmetry. Face 1 is taken to be one of these straight edges, face 2 is the inner circle arc and face 3 is the outer circle arc. The shape of an annulus is completely determined by the ratio of the outer to inner radius, and so to perform a

parametric study, the inner radius was fixed to be equal to 1, and the outer radius was taken to be $1+\alpha$, with α varying from 0.025 to 0.25.

The results of this study are presented in Figure 12. For face 1, the literature method and stiffness matrix method give fairly similar TICs. However, the TICs calculated by the mass matrix method are much larger than both, and so this is assumed to be a coincidence. The results for face 2 and face 3 are qualitatively similar, which is to be expected as they are both circle arcs, just with different radii. In both cases, the mass matrix and literature methods produce similar TICs, while the stiffness matrix TICs are a factor of 1.69 to 2.25 lower than those calculated by the literature method.

For the moderator element, all four faces are unique. Face 2 is taken to be the circle arc that is shared with a cladding element, face 3 is the edge opposite face 2 that coincides with the domain boundary, face 4 is the diagonal edge that separates the two moderator elements, and face 1 is the edge on the domain boundary that is opposite face 4. The shape of the moderator patch can also be uniquely defined by a single parameter, by taking the square domain in Figure 11 to be of side length 1, and then varying the outer radius of the cladding annulus by a parameter α. α was varied between 0.1 and 0.9 in this study.



(a) Face 1

(b) Face 2

(c) Face 3

(d) Face 4

Figure 13: The TICs for the faces of a moderator patch as shown in Figure 11, as the outer radius of the cladding annulus is varied inside a fixed square geometry of side length 1.

The results of this study are presented in Figure 13. The behaviour of the stiffness matrix method compared to the other two methods is very different for all four faces of a moderator patch. For face 3, for example, the TICs

calculated using the stiffness matrix method are consistently lower than those calculated by the method from the literature. However, the opposite is true of face 1, and so penalty parameters computed using the literature method would not guarantee coercivity of the bilinear form. In this example it can also be seen that for more complex element shapes, the relative values of the TICs computed by the three methods are correspondingly more complicated. As with the quarter circle, this serves to strongly suggest that the full inequality of interest involving the gradient in Equation 32 must be taken into account when dealing with NURBS.

4.3. Method of Manufactured Solutions

The numerical results presented so far highlight the differences between the TICs calculated by the three different methods. However, it is the difference in the resulting penalty parameters that impact the solution of the model problem presented in Section 2.1. In order to assess this impact, the method of manufactured solutions will be used to create the source term $Q(\mathbf{r})$ in Equation 1, for a selection of constant material properties D and Σ_a.

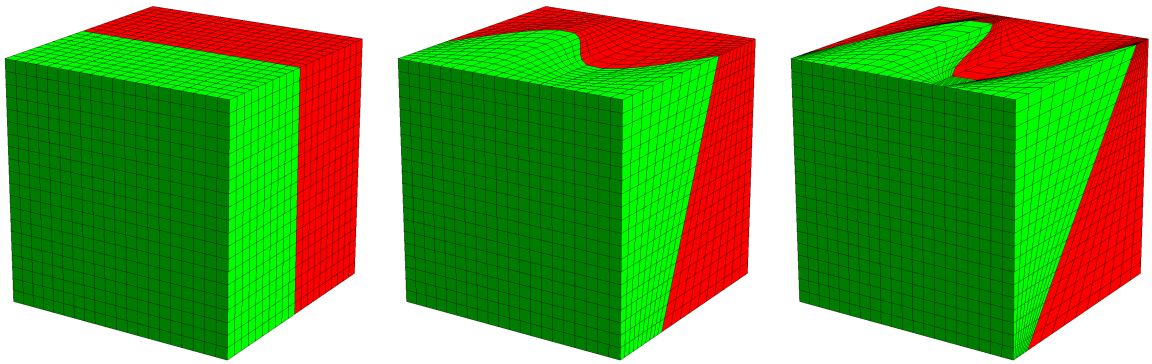
The domain is given by a unit centimetre cube $[0, 1]^3$, and the manufactured solution $\phi_m(\mathbf{r})$ is given by:

$$\phi_m(\mathbf{r}) = (1-x)^2(1-y)^2(1-z)^2 \tag{43}$$

$$\implies Q(\mathbf{r}) = -D\nabla^2\phi_m(\mathbf{r}) + \Sigma_a\phi_m(\mathbf{r}) \tag{44}$$

as D and Σ_a are now constant. This leads to homogeneous boundary conditions, which are Neumann on the $x = 0$, $y = 0$ and $z = 0$ faces, and Dirichlet on the $x = 1$, $y = 1$ and $z = 1$ faces.

To create sequences of linear hexahedral meshes containing elements with varying levels of distortion, the domain was split into two regions through the plane $x = 0.5$, as shown in Figure 14a. The bounding curves in the $z = 0$ and $z = 1$ planes of the surface separating the two regions were then distorted into B-splines, as shown in Figures 14b and 14c. These curves were then interpolated between to form the surface separating the two regions, and these regions were then meshed in a structured manner with four different levels of refinement.



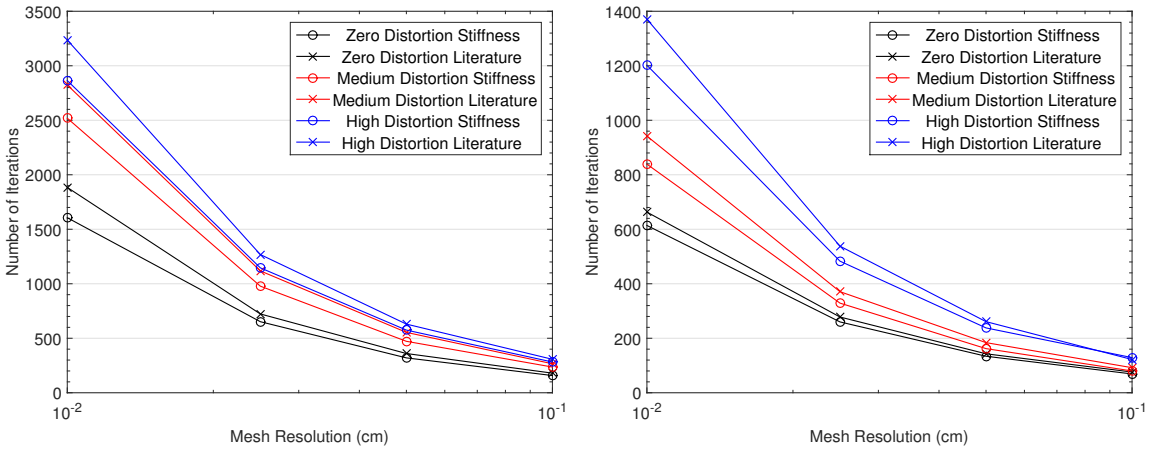
(a) Zero distortion (b) Medium distortion (c) High distortion

Figure 14: Varying levels of mesh distortion for the MMS problem over the unit cube.

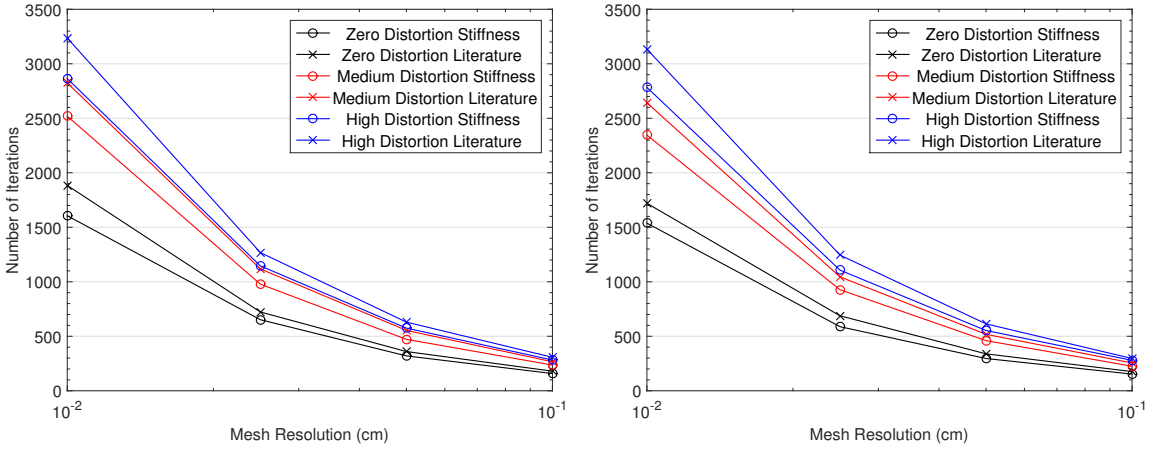
For the material constants, values were selected based on typical radiation/heat transfer problems. Both high ($D = 33.\dot{3}$ cm) and low ($D = 0.0\dot{3}$ cm) diffusivity problems were considered. For each diffusion coefficient, three Σ_a values were used, for a total of six test cases. From high to low, these Σ_a values represent strongly absorbing, highly scattering and purely scattering materials respectively.

These MMS problems were then solved using the SIP scheme defined in Equation 9. The penalty parameters from Equations 20 and 21 were calculated using both TICs from the literature, defined in Equation 23, and those calculated using the stiffness matrix method.

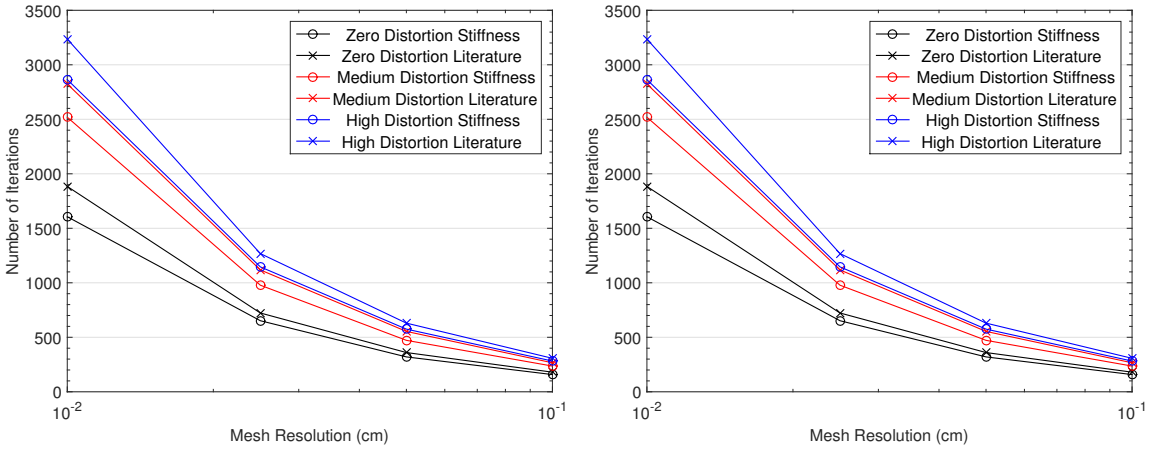
Comparisons were first made based on the L_2 error compared to the manufactured solution $\phi_m(\mathbf{r})$. For all six test cases, and for all three mesh distortion levels, the SIP scheme using penalty parameters computed by both methods



(a) $D = 33.\dot{3}$ cm, $\Sigma_a = 0.009$ cm^{-1} (b) $D = 0.0\dot{3}$ cm, $\Sigma_a = 9.0$ cm^{-1}



(c) $D = 33.\dot{3}$ cm, $\Sigma_a = 0.0001$ cm^{-1} (d) $D = 0.0\dot{3}$ cm, $\Sigma_a = 0.1$ cm^{-1}



(e) $D = 33.\dot{3}$ cm, $\Sigma_a = 0.0$ cm^{-1} (f) $D = 0.0\dot{3}$ cm, $\Sigma_a = 0.0$ cm^{-1}

Figure 15: The number of block-Jacobi preconditioned conjugate gradients iterations required for convergence of the linear system for the MMS problem. The six different test cases are presented in separate subfigures. High diffusivity problems are on the left, low diffusivity problems are on the right, and the absorption is decreasing from top to bottom. Each subfigure contains all three mesh distortion levels, and both penalty parameter calculation methods, as the mesh is refined.

achieved second order convergence as the mesh was refined. In addition, for a fixed test case and mesh, the difference between the L_2 error computed using the penalty parameters based on the two different TIC calculation methods was negligible. Plots of these results have been omitted for brevity.

Comparisons were then made between the two penalty parameter calculation methods based on the efficiency of solving the resulting linear system. In all cases, the conjugate gradient method is used to solve the linear system, with an element level block-Jacobi preconditioner. The total number of iterations required to reduce the norm of the residual divided by the norm of the right hand side vector to 10^{-12} is then used as a metric for the linear system solution efficiency. The results of this comparison for all six test cases and all three mesh distortion levels are shown in Figure 15.

We note that while there are significantly more efficient preconditioners available, their use has not been investigated here, in order to facilitate efficiency comparisons between the various TICs. For example, using an algebraic multigrid precondioner drastically reduces the iteration count, from $\sim 10^3$ in Figure 15, to $\sim 10^1$. This makes it difficult to accurately ascertain the impact of the TICs on solution efficiency, as each iteration represents a much larger percentage of the total iteration count.

For every test case, mesh distortion level, and mesh refinement level, the linear system required fewer iterations to solve when the penalty parameters were calculated using the stiffness matrix method compared to the literature method, except for one. When $D = 0.0\dot{3}$ cm, $\Sigma_a = 9.0$ cm^{-1}, and the coarsest, most distorted mesh is used, the linear system required slightly more iterations to solve using the stiffness matrix method compared to the literature method. As this is precisely the mesh in which the elements are the most distorted, it is hypothesised that the literature method has produced lower penalty parameters than the stiffness matrix method in this case, in which case coercivity is not guaranteed.

Aside from this data point, the stiffness matrix method produces savings in the number of iterations of $5.7 - 14.8\%$ for the test cases considered, and an average saving of 11.0%. As the operations performed at each iteration are identical, these savings are also achieved in the total solution time of the linear system.

5. Conclusion

In all applications of incomplete or symmetric interior penalty methods, the penalty parameters on element faces must be sufficiently large to guarantee that the resulting bilinear form is coercive. As the condition number, and therefore solution time, of the resulting matrix increase as the penalty parameters increase, it is desirable to choose the minimum penalty parameters possible that still guarantee coercivity of the bilinear form. Their derivation rely on trace inequalities, and the resulting penalty parameters are proportional to the constants $C_{e,f}$ that appear in these trace inequalities.

To calculate these constants, current methods in the literature focus on finding closed-form expressions for specific element types. However, these methods bound integrals of functions from the finite element function space, when in the penalty parameter derivation it is the integrals of gradients of these functions that are important, and therefore some information is discarded that could be used to improve (lower) the trace inequality constants. In addition, these methods are only strictly valid if the mapping from the reference element to the physical element is linear, which is only guaranteed to be the case for linear simplices.

Rather than relying on such closed-form expressions, which must be derived separately for each new element type, we propose to calculate the TICs numerically for each face of each element. The method presented to do so involves the numerical solution of a generalised eigenvalue problem involving the volumetric and face stiffness matrices, and this has a number of advantages over the existing methods in the literature. Firstly, the mapping from the reference element to the physical element is inherently taken into account in the stiffness matrices, and therefore the method can be applied to elements that do not have a constant Jacobian. Secondly, the method explicitly takes into account that the penalty parameter derivation depends on the integrals of gradients of functions from the finite element function space, and so no information is discarded in their calculation. Thirdly, all of the relevant information required for calculating the TICs is completely contained in the volumetric and face stiffness matrices. The method is therefore applicable to arbitrary element types for which these matrices can be computed, including, but not restricted to, finite elements and the NURBS used in IGA. Finally, the method is provably sharp for the function spaces of interest, and so the lowest possible penalty parameters can be computed by this method.

As the generalised eigenvalue problem is singular, some care must be taken in its solution. A robust and practical method has been presented for doing so when the associated matrices are relatively small, such as those based on discretisations in which the basis functions are discontinuous between every element. For problems with large continuous regions, such as multipatch IGA, in which these matrices may have dimension $O(10^6)$, the method presented would be extremely computationally demanding, and in this case, some suggestions have been made as to how such problems may be approached.

Numerical results for elements with constant Jacobians demonstrated two important properties of the stiffness matrix method compared to the existing methods in the literature. The TICs, and therefore associated penalty parameters, computed using the stiffness matrix method are always lower than those computed by the methods from the literature, by up to a factor of 4 in the most extreme example. This was expected, as the methods from the literature provide sharp bounds on the integrals of functions from the finite element space, of which the gradients of functions are members in the constant Jacobian case. The extra saving is then realised in the stiffness matrix method by explicitly bounding the gradients of the finite element functions.

In addition, the literature methods take into account the different faces of a given element only through the area of the face $\mathcal{A}(f)$. However, the stiffness matrix method accounts directly for anisotropy in the element shape, and so provides TICs that are more specifically tailored to a particular face of an element. As penalty parameters for a face are proportional to the TICs, this should directly translate into improved penalty parameters for IP methods.

A variety of finite elements with non-constant Jacobians were also used to compare the various methods of TIC calculation. Although the methods from the literature are not strictly valid for these elements, in practice, these methods [15], or methods that are extremely closely related [13, 26], are used in practice. In this case, the TICs computed using the methods from the literature were frequently lower than those computed by the stiffness matrix method. As the stiffness matrix method is provably sharp, the bilinear form using the penalty parameters computed using the TICs from the literature would therefore not be coercive in general, and large errors would be introduced into the resulting numerical solution.

In order to demonstrate the application of the method to IGA, a quarter of a pincell geometry was constructed using biquadratic NURBS patches, as this is a frequently arising geometry in reactor physics applications. In this case, the increased complexity of the element shapes, as shown in Figure 11, drive a corresponding increase in complexity of the TICs as the relative sizes of the elements vary. In particular, for the circle arc faces of the quarter circle, the TICs computed by the stiffness matrix method are $\times 50$ larger than those calculated by the methods from the literature. This demonstrates that for general IGA applications, heuristic approaches based on such methods can fail drastically even for relatively simple element shapes.

The final test case involved a manufactured solution of the model reaction-diffusion equation presented in Section 2.1. A variety of material properties and distorted meshes constructed from linear hexahedra were considered. Penalty parameters were calculated using both the stiffness matrix method, and the method from the literature for these elements, and the solution times of the iterative method used to solve the resulting linear system compared. The stiffness matrix method was found to reduce the solution time by approximately 11% compared to the literature method on average for the range of test cases considered.

Standard finite element codes solving the model reaction-diffusion equation from Section 2.1 by interior penalty methods must already compute the volumetric stiffness matrix for each element, and would require only a small modification to compute the face stiffness matrices as well. For discretisations in which the basis functions are discontinuous between every element, the numerical solution of the generalised eigenvalue problems is expected to be much less computationally expensive than the solution of the resulting linear system. It is noted that these TIC calculations are trivially parallelisable, as they are local to each element, and that the face stiffness matrices can be discarded once the $C_{e,f}$ have been calculated, minimising the memory overhead.

For elements with a constant Jacobian, the worst case scenario is that the presented method produces the same TICs, and therefore penalty parameters as the literature, while the best case scenario is a saving of a factor of $\times 4$, and these savings translate directly into savings in the solution time of the resulting linear system. For elements with a non-constant Jacobian, and NURBS in particular, the authors are not aware of any other method to reliably compute penalty parameters that guarantee coercivity of the bilinear form, and so the presented method should be useful in a wide variety of applications.

Acknowledgements

The authors would like to acknowledge Richard Smedley-Stevenson for his valuable discussions on the topics of IP methods and DSA. A.R. Owens would like to acknowledge the support of EPSRC under their industrial doctorate programme (EPSRC grant number: EP/G037426/1), Rolls-Royce for industrial support and the Imperial College High Performance Computing Service for technical support. M.D. Eaton and J. Kópházi would like to would like to thank EPSRC for their support through the following grants: "Adaptive Hierarchical Radiation Transport Methods to Meet Future Challenges in Reactor Physics" (EPSRC grant number: EP/J002011/1) and "RADIANT: A Parallel, Scalable, High Performance Radiation Transport Modelling and Simulation Framework for Reactor Physics, Nuclear Criticality Safety Assessment and Radiation Shielding Analyses" (EPSRC grant number: EP/K503733/1). In accordance with EPSRC funding requirements, all supporting data used to create figures and tables in this paper may be accessed at the following URL: https://doi.org/10.5281/zenodo.580072.

Appendix A. Implementation overview

This section is intended to serve as a "how to" guide for the practical implementation of penalty parameters based on the trace inequalities presented in this paper, and is for the IP practitioner who is not concerned with the origin of the penalty parameters, only with their effectiveness. As such it will be presented as an algorithm, which will reference formulae from the main text where appropriate.

Algorithm 1 Implementation algorithm

1: **for** $e = 1$ **to** $N_{elements}$ **do**
2: Calculate the $N \times N$ element volumetric stiffness matrix $\underline{S}_v$ as defined in Equation 36b
3: Calculate the vector (of length N) $\mathbf{u}_c$ that defines the constant function over this element, and normalise it
4: **for** $f = 1$ **to** $N_{faces}(e)$ **do**
5: Calculate the $N \times N$ face stiffness matrix $\underline{S}_f$ as defined in Equation 36d
6: **if** N is sufficiently small **then**
7: Create an orthonormal basis of $\mathbb{R}^N$ that contains the vector $\mathbf{u}_c$ (e.g. by modified Gram-Schmidt)
8: Form the $N \times N - 1$ matrix $\underline{X}$, whose columns contain all of these orthonormal basis vectors except $\mathbf{u}_c$
9: Orthogonalise out $\mathbf{u}_c$ by the transformations $\underline{S}'_v = \underline{X}^T \underline{S}_v \underline{X}$ and $\underline{S}'_f = \underline{X}^T \underline{S}_f \underline{X}$
10: Calculate the maximum generalised eigenvalue λ_{max} of the system $\left(\underline{S}'_f - \lambda \underline{S}'_v\right)\mathbf{x} = \mathbf{0}$
11: **else if** N is large **then**
12: Calculate the maximum generalised eigenvalue λ_{max} of the system $\left(\underline{S}_f - \lambda \underline{S}_v\right)\mathbf{x} = \mathbf{0}$
13: **end if**
14: Set the trace inequality constant for this face of this element $C_{e,f} = \lambda_{max}$
15: **end for**
16: **end for**
17: **for** $f = 1$ **to** N_{faces} **do**
18: Compute the penalty parameter for face f by $\mu_f = \frac{(1+\theta)^2}{8} \max_{e \ni f} \left\{ D^e C_{e,f} N_e \right\}$
19: **end for**

The key points of each line of the algorithm will now be covered:

Line 1: $N_{elements}$ is the number of elements in the mesh, the number of patches in IGA, or the number of regions between which the solution can be discontinuous in general.

Line 2: N is the number of basis functions of element e.

Line 3: In general, this vector can be computed by Galerkin projection, and its calculation is discussed in more detail in Section 3. The kernel of $\underline{S}_v$ is spanned by this vector, and so is the shared kernel of $\underline{S}_v$ and $\underline{S}_f$.

Line 4: $N_{faces}(e)$ is the number of faces of element e.

Line 6: The two cases that this *if* statement switch between will calculate the same value of $C_{e,f}$, but the method that is most efficient will depend on the size of N.

Line 7: Modified Gram-Schmidt was used here, although any method that achieves an orthonormal basis is equally applicable, e.g. Householder transformations.

Line 8: $\underline{X}$ is the transformation matrix to the $N-1$ dimensional vector space $\mathbb{R}^N \setminus \text{span}\{\mathbf{u}_c\}$.

Line 9: With the shared kernel eliminated, $\underline{S}'_v$ is SPD, which greatly simplifies the calculation of λ_{max}.

Line 10: The LAPACK routine DSYGV [22] was utilised in this work, although a power iteration could also be used.

Line 12: For large, sparse systems, the explicit orthogonalisation against the shared kernel, spanned by $\mathbf{u}_c$, described above, is extremely computationally expensive. An alternative is to perform a power iteration. This is complicated by the fact the $\underline{S}_v$ is singular, but the knowledge of the kernel should make the application of projected Krylov methods [23–25] feasible.

Line 17: N_{faces} now defines the number of faces in the mesh. Internal faces are defined to be $\partial V_e^i \cap \partial V_e^j$ for some pair of elements with indices i and j, i.e. the face separates precisely two elements (which is important when a mesh contains hanging-nodes). Boundary faces are defined naturally as the intersection of the element faces with the domain boundary.

Line 18: D^e is the diffusion coefficient in element e, N_e is the number of faces of element e that are not on a Neumann boundary, and θ is equal to zero for the IIP scheme and one for the SIP scheme. The maximum is taken over the (at most two) elements e that share the face f.

References

[1] Joachim Nitsche. Über ein variationsprinzip zur lösung von dirichlet-problemen bei verwendung von teilräumen, die keinen randbedingungen unterworfen sind. In *Abhandlungen aus dem mathematischen Seminar der Universität Hamburg*, volume 36, pages 9–15. Springer, 1971.

[2] Mary Fanett Wheeler. An elliptic collocation-finite element method with interior penalties. *SIAM Journal on Numerical Analysis*, 15:152–161, 1978.

[3] Douglas N. Arnold. An interior penalty finite element method with discontinuous elements. *SIAM Journal on Numerical Analysis*, 19:742–760, 1982.

[4] Roland Becker, Peter Hansbo, and Rolf Stenberg. A finite element method for domain decomposition with non-matching grids. *Mathematical Modelling and Numerical Analysis*, 37:209–225, 2003.

[5] Garth A. Baker, Wadi N. Jureidini, and Ohannes A. Karakashian. Piecewise solenoidal vector fields and the Stokes problem. *SIAM Journal on Numerical Analysis*, 27:1466–1485, 1990.

[6] M. Drosson and K. Hillewaert. On the stability of the symmetric interior penalty method for the Spalart-Allmaras turbulence model. *Journal of Computational and Applied Mathematics*, 246:122–135, 2013.

[7] Béatrice Rivière and Vivette Girault. Discontinuous finite element methods for incompressible flows on subdomains with non-matching interfaces. *Computer methods in applied mechanics and engineering*, 195:3274–3292, 2006.

[8] Yekaterina Epshteyn and Béatrice Rivière. Estimation of penalty parameters for symmetric interior penalty Galerkin methods. *Journal of Computational and Applied Mathematics*, 206:843–872, 2007.

[9] F. Brunero. Discontinuous Galerkin methods for isogeometric analysis. Master's thesis, Università degli Studi di Milano, 2012.

[10] Christoph Hofer and Ulrich Langer. Dual-primal isogeometric tearing and interconnecting solvers for multipatch dG-IgA equations. *Computer methods in applied mechanics and engineering*, 316:2–21, 2017.

[11] Khosro Shahbazi. An explicit expression for the penalty parameter of the interior penalty method. *Journal of Computational Physics*, 205:401–407, 2005.

[12] Paul Castillo. Performance of discontinuous Galerkin methods for elliptic PDEs. *SIAM Journal of Scientific Computing*, 24:524–547, 2002.

[13] Yaqi Wang and Jean C. Ragusa. Diffusion Synthetic Acceleration for High-Order Discontinuous Finite Element S_N Transport Schemes and Application to Locally Refined Unstructured Meshes. *Nuclear Science and Engineering*, 166:145–166, 2010.

[14] T. Warburton and J.S. Hesthaven. On the constants in hp-finite element trace inverse inequalities. *Computer Methods in Applied Mechanical Engineering*, 192:2765–2773, 2003.

[15] Koen Hillewaert. *Development of the discontinuous Galerkin method for high-resolution, large scale CFD and acoustics in industrial geometries*. PhD thesis, Université Catholique de Louvain, 2013.

[16] John A. Evans and Thomas J.R. Hughes. Explicit trace inequalities for isogeometric analysis and parametric hexahedral finite elements. *Numerische Mathematik*, 123:259–290, 2013.

[17] H.G. Stone and M.L. Adams. A piecewise linear finite element basis with application to particle transport. In *Proc. ANS Topical Meeting Nuclear Mathematical and Computational Sciences Meeting*, pages 6–11, 2003.

[18] T.S. Bailey. *A piecewise linear discontinuous finite element method applied to the RZ and XYZ transport equations*. PhD thesis, Texas A& M University, 2008.

[19] D. A. Di Pietro and A. Ern. Pdes with diffusion. In D. A. Di Pietro and A. Ern, editors, *Mathematical Aspects of Discontinuous Galerkin Methods*, chapter 4, pages 119–186. Springer-Verlag Berlin Heidelberg, 2012.

[20] Isaac Harari and Thomas J.R. Hughes. What are C and h?: Inequalities for the analysis and design of finite element methods. *Computer Methods in Applied Mechanics and Engineering*, 97:157–192, 1992.

[21] Yousef Saad. *Numerical Methods for Large Eigenvalue Problems*. Society for Industrial and Applied Mathematics, 2011.
[22] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen. *LAPACK Users' Guide*. Society for Industrial and Applied Mathematics, Philadelphia, PA, third edition, 1999.
[23] D. Orban. Projected Krylov methods for unsymmetric augmented systems. Technical Report G-2008-46, GERAD, 2008.
[24] N. Gould, D. Orban, and T. Rees. Projected Krylov methods for saddle-point systems. *SIAM Journal on Matrix Analysis and Applications*, 35:1329–1343, 2014.
[25] R. Kucera, T. Kozubek, A. Markopoulos, J. Haslinger, and L. Mocek. Projected Krylov methods for solving non-symmetric two-by-two block linear systems arising from fictitious domain formulations. *Advances in Electrical and Electronic Engineering*, 12:131–143, 2014.
[26] B. Turcksin and J.C. Ragusa. Discontinuous diffusion synthetic acceleration for S_N transport on 2D arbitrary polygonal meshes. *Journal of Computational Physics*, 274:356–369, 2014.