

# ICES REPORT 16-11

---

May 2016

## Optimal and reduced quadrature rules for tensor product and hierarchically refined splines in isogeometric analysis

by

René R. Hiemstra, Francesco Calabrò, Dominik Schillinger, and Thomas J.R. Hughes



**The Institute for Computational Engineering and Sciences**  
The University of Texas at Austin  
Austin, Texas 78712

*Reference: René R. Hiemstra, Francesco Calabrò, Dominik Schillinger, and Thomas J.R. Hughes, "Optimal and reduced quadrature rules for tensor product and hierarchically refined splines in isogeometric analysis," ICES REPORT 16-11, The Institute for Computational Engineering and Sciences, The University of Texas at Austin, May 2016.*

# Optimal and reduced quadrature rules for tensor product and hierarchically refined splines in isogeometric analysis

René R. Hiemstra<sup>a\*</sup>, Francesco Calabrò<sup>b</sup>, Dominik Schillinger<sup>c</sup>  
& Thomas J.R. Hughes<sup>a</sup>

<sup>a</sup> *Institute for Computational Engineering and Sciences, The University of Texas at Austin,*

<sup>b</sup> *DIEI, Università di Cassino e del Lazio Meridionale,*

<sup>c</sup> *Department of Civil, Environmental, and Geo-Engineering, University of Minnesota.*

\* Corresponding author. *E-mail address: rene@ices.utexas.edu*

---

## Abstract

We continue the study initiated in [29] in search of optimal quadrature rules for tensor product and hierarchically refined splines in isogeometric analysis. These rules are optimal in the sense that there exists no other quadrature rule that can exactly integrate the elements of the given spline space with fewer quadrature points. We extend the algorithm presented in [29] with an improved starting guess, which combined with arbitrary precision arithmetic, results in the practical computation of quadrature rules for univariate non-uniform splines up to any precision. Explicit constructions are provided in sixteen digits of accuracy for some of the most commonly used uniform spline spaces defined by open knot vectors. We study the efficacy of the proposed rules in the context of full and reduced quadrature applied to two- and three-dimensional diffusion-reaction problems using tensor product and hierarchically refined splines, and prove a theorem rigorously establishing the stability and accuracy of the reduced rules.

*Keywords:* Isogeometric analysis; Optimal quadrature rules; Generalized Gaussian quadrature rules; Reduced quadrature rules

---

# 1 Introduction

Isogeometric analysis [17, 26] is a relatively new computational mechanics technology which aims to unify the historically separated fields of engineering analysis and computer aided design (CAD). It appears that isogeometric analysis offers several important advantages over classical finite element analysis, such as increased robustness and accuracy on a per degree-of-freedom basis [17, 26, 35], and improved spectral properties [18, 19, 28], important in hyperbolic problems. However, much work remains to be done when it comes to efficient implementation. Inspiration can be drawn from 50 years of experience with the classical finite element method. However, the structure of isogeometric stiffness and mass matrices is different and requires novel efficient numerical linear solvers, pre-conditioners, and domain decomposition methods for parallel implementations. The topic of this paper also falls into this category, namely efficient numerical quadrature.

Quadrature is necessary to numerically evaluate finite element mass and stiffness matrices. Since the finite element matrix formation and assembly is responsible for up to half of the total computational time in many isogeometric analysis problems, it is worthwhile to design more efficient quadrature rules. As demonstrated by Hughes, Reali, and Sangalli in [29], standard quadrature rules, such as Gauss-Legendre, can be far from optimal because they do not take into account the increased regularity of isogeometric function spaces. By carefully considering the polynomial degree and the regularity, the asymptotic minimum number of quadrature points and weights can be determined that are required for exact integration. Tables 1 and 2 show the difference as compared with Gauss-Legendre quadrature. One can imagine that in two- and three-dimensional problems the savings can be significant for increased continuity function spaces.

Table 1: Asymptotic number of quadrature points per element,  $m/n_{\text{el}}$ , for integrating  $n_{\text{sd}}$ -dimensional tensor product splines of degree  $q$  and regularity  $r$ , meaning  $r$  is the number of continuous derivatives.

	Gauss Legendre quadrature	Optimal quadrature
$m/n_{\text{el}}$	$\lceil \frac{q+1}{2} \rceil^{n_{\text{sd}}}$	$\left(\frac{q-r}{2}\right)^{n_{\text{sd}}}$

## 1.1 Generalized Gaussian quadrature

A quadrature rule is a numerical approximation to a definite integral of a function  $\phi(x)$ , usually in terms of a weighted linear combination of function evaluations at specified points in the domain of integration. The rule thus corresponds to a set of quadrature points and weights  $\{(\bar{\xi}_j, \bar{w}_j) : j = 1, \dots, m\}$ , such that the quadrature formula,

$$\sum_{j=1}^m \bar{w}_j \phi(\bar{\xi}_j) \approx \int_a^b \phi(\xi) d\xi, \quad (1)$$

is sufficiently accurate for all functions  $\phi$  in some pre-determined function space  $\mathcal{V}$ . Typically, the formula is required to be exact for an  $n$ -dimensional linear subspace  $\mathcal{V}^n \subset \mathcal{V}$ , called the

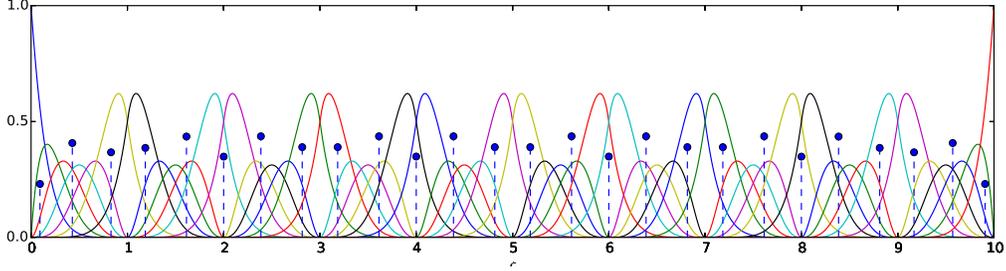


Figure 1: Distribution of 26 quadrature points and corresponding weights (vertical coordinate) for integrating the 52 B-spline functions in the space  $\mathcal{S}_1^6(0, 10, 10)$ , that is, the space of sixth degree piecewise polynomials having a continuous derivative on the ten element domain  $(0, 10)$ .

target space. An important observation here is that the quadrature formula in (1) has  $2m$  free parameters:  $m$  points and  $m$  weights, so we can hope to integrate exactly all functions in a target space of dimension  $n = 2m$ . For example, an  $m$ -point Gauss-Legendre rule exactly integrates all  $2m$  polynomials  $x^k, k = 0, 1, \dots, 2m - 1$ , and is optimal in the sense that there exists no other rule that exactly integrates all polynomials with fewer quadrature points and weights.

In the general setting, if  $\{\phi_i, i = 1, \dots, 2m\}$  is a basis for the target space  $\mathcal{V}^{2m}$ , then the  $m$  quadrature points and  $m$  weights  $\{(\bar{\xi}_j, \bar{w}_j) : j = 1, \dots, m\}$  are given as the unique solution - if it exists - of the non-linear system of  $2m$  equations,

$$\begin{aligned} \sum_{j=1}^m \phi_1(\bar{\xi}_j) \bar{w}_j &= \int_a^b \phi_1(\xi) d\xi \\ &\vdots \\ \sum_{j=1}^m \phi_{2m}(\bar{\xi}_j) \bar{w}_j &= \int_a^b \phi_{2m}(\xi) d\xi. \end{aligned} \tag{2}$$

Such rules are called generalized Gaussian quadrature rules, referring to the fact that there exists no other rule that exactly integrates all  $\phi_i$  in the target space  $\mathcal{V}^{2m}$  with fewer points and weights. In the sequel of this paper we will refer to these rules simply as Gaussian or optimal quadrature rules. The conditions for existence and uniqueness of Gaussian rules have been described by Karlin in [33] and algorithms for their numerical computation have been presented in [13, 15, 36].

An example of a Gaussian quadrature rule for a  $C^1$  spline space of degree six defined on a uniform open knot vector of ten elements is shown in Figure 1. This spline space has 52 basis functions, which can be exactly integrated with 26 quadrature points and corresponding weights. Gaussian quadrature for splines has been studied in [32, 39, 45]; see also [46]. In an early paper, [45], Schoenberg postulated the existence and uniqueness of Gaussian quadrature rules for splines. These claims were later proved in simple cases by Karlin and Micchelli [32, 39]. To date, however, no proof of existence and uniqueness of spline Gaussian quadrature rules exist for the case of arbitrary degree and knot vector.

## 1.2 Related work

Hughes, Reali and Sangalli [29] commenced the study of optimal quadrature in Galerkin based tensor product isogeometric analysis. A Newton method following [36, 15] was presented in order to solve the system of non-linear equations in (2), and several rules of practical interest were obtained. However, these were limited to spline spaces of modest degree, continuity, and dimension due to the highly non-linear behavior of the equations. Furthermore, ill-conditioning limited the accuracy of the rules below the preferred sixteen digits of accuracy.

Some of these difficulties have been addressed in recent work by Ait-Haddou, Bartoň, and Calo in [4, 7, 8, 9]. In [4] and [7] progress has been made on the space of non-uniform  $C^1$  splines of degree 3 and uniform  $C^1$  splines of degree 5 using analytical techniques based on previous work in [40]. Recently, the same authors developed a new methodology in which a known quadrature formula is transformed from a source space to the target space by continuous deformation of the knot vector. Instead of determining a sufficiently accurate starting guess, the method requires a source space with known quadrature formula that is reasonably *close* to the target space. The rule is updated iteratively by means of a Newton method based on polynomial homotopy continuation. In [9] this approach is outlined for  $C^2$  non-uniform cubic splines, and in [8] the authors extend this work to odd degree splines of variable regularity and open knot vector. Some explicit quadrature formulae of practical interest were presented for uniform spline spaces defined on an open knot vector.

Similar in scope is the work by Johannessen [30], who presented a practical algorithm based on [29] and provided an accompanying Matlab implementation. Robust results were obtained using an improved starting guess, and, inspired by [9], homotopy continuation of the knot vector. Rules were computed for a range of target spline spaces defined on a 128 element partition. Some cases appeared to require a large number of iterations compared to others. Furthermore, ill conditioning of the non-linear equations limited the accuracy of the rules to approximately ten digits.

Further contributions that exploit the increased continuity of splines in quadrature include those of [6] and [44]. In [6] a scheme was proposed based on solving a series of local non-linear optimization problems. Although the resulting quadrature rules are close to optimal, the scheme works only in the case of a uniform partition and is complicated by the fact that it does not directly utilize the B-splines as a basis for computation. A particularly simple approach to efficient quadrature in the non-uniform setting was proposed by Schillinger, Hossain, and Hughes in [44], where  $C^0$  rules, such as Gauss-Lobatto and certain monomial rules, were studied in the context of exact and reduced integration. Substantial savings were obtained by sharing quadrature points on the boundary interface between neighboring elements, in particular for low order finite elements in three dimensions.

Other developments regarding quadrature in isogeometric analysis include those of [1, 2, 3, 24, 31, 37, 38]. In [31] GPU based parallel computation of the isogeometric stiffness matrix was investigated. Different matrix assembly strategies were discussed in [38]. In [37] a different approach to quadrature based on interpolation and table lookup was proposed. In [1, 2, 3] selective and reduced quadrature schemes were proposed to counteract membrane and transverse shear locking in isogeometric beam, plate, and shell analysis. Furthermore, in [24] stable and accurate reduced quadrature was studied in the context of variationally

consistent domain integration. We refer the reader to these papers for more information.

**Remark 1.1.** *Most of these contributions, including the present study, are aimed toward the construction of quadrature rules for univariate splines on an entire macro-element or patch. Nevertheless, the quadrature points and corresponding weights specific to a particular Bézier element can easily be identified, and hence, their application is straightforward in standard element-based isogeometric analysis. Furthermore, due to the underlying tensor product structure, these quadrature rules are well suited for application in a nested sum-factorization [5], saving yet another 50% on the number of operations required in the assembly process.*

### 1.3 Contributions of this work

We continue the study initiated in [29] in search of optimal quadrature rules for tensor product NURBS and hierarchically refined splines in isogeometric analysis. Our approach to computing quadrature rules is based on the same non-linear equations as proposed in [29]. Similar to [30] we provide an improved starting guess and use a continuation algorithm in order to consistently produce accurate results. The proposed starting guess, obtained by solving a series of localized non-linear problems, is highly accurate and in certain cases exact. Our approach to continuation is very simple and is based on the residual of the the non-linear equations in (2). The resulting quadrature rules are computed to 16 digits of accuracy by employing arbitrary precision arithmetic.

We present explicit quadrature formulae in 16 digits of accuracy for some of the more commonly used uniform spline spaces defined on an open knot vector of degree and regularity  $(q, r)$ :

	even degree	odd degree
$p = 2, 3, 4, 5$	$(2p, p - 2)$	$(2p - 1, p - 2)$

Table 2 illustrates the possible savings of these rules for use in isogeometric analysis. Even degree rules provide full integration of isogeometric mass and stiffness matrices, with significantly fewer quadrature points, up to a factor of five in 3D, as compared with Gauss-Legendre quadrature. The odd degree rules can be effectively used in the context of reduced integration, providing another factor of two in computational savings. The rules corresponding to  $(3, 0)$  and  $(4, 0)$  have been previously determined in [29], and for the space  $(5, 1)$  in [7]. To our knowledge this is the first work in which an explicit rule for the space  $(6, 1)$  is introduced. This rule has practical value because it provides full integration of the mass and stiffness matrix in cubic, maximally smooth, isogeometric analysis, for second order operators, such as arise in potential theory and elasticity.

A major focus of our work, distinguishing our contribution from others, is the application of the computed rules in the context of exact and reduced integration applied to two and three dimensional boundary value problems. In particular, we prove a theorem rigorously establishing the stability and accuracy of the reduced rules applied to elliptic problems with sufficiently smooth data. Finally, we illustrate the use of optimal quadrature in the

Table 2: Comparison of asymptotic number of quadrature points per Bézier element for Gauss-Legendre, optimal and reduced rules. We consider tensor product isogeometric analysis with trial space of polynomial degree  $p$  and regularity  $p - 1$ . The integrands of the bi-linear forms under consideration are of degree  $2p$  and regularity  $p - 2$ .

(degree, regularity)		Full Gauss-Legendre			Full optimal			Reduced optimal		
Trial space	Integrand	1D	2D	3D	1D	2D	3D	1D	2D	3D
(2,1)	(4,0)	3.00	9.00	27.0	2.00	4.00	8.00	1.50	2.25	3.38
(3,2)	(6,1)	4.00	16.0	64.0	2.50	6.25	15.6	2.00	4.00	8.00
(4,3)	(8,2)	5.00	25.0	125	3.00	9.00	27.0	2.50	6.25	15.6
(5,4)	(10,3)	6.00	36.0	216	3.50	12.3	42.9	3.00	9.00	27.0

context of isogeometric analysis based on local refinement through hierarchical B-splines [12, 22, 34, 42, 43, 50].

## 1.4 Outline

In Section 2 we discuss a selection of techniques to compute optimal quadrature rules for univariate non-uniform splines. Furthermore, explicit rules for some of the more commonly used uniform spline spaces are presented. In Section 3 we study the efficacy of the proposed full and reduced rules applied to two- and three-dimensional diffusion-reaction problems using tensor product NURBS and hierarchically refined splines. In Section 4 we prove a theorem establishing stability and accuracy of the reduced quadrature rules. Finally, in Section 5 we evaluate our contributions and recommend directions for future work.

## 2 Optimal quadrature formulae for univariate splines

In this section we discuss a combination of techniques that result in the robust and accurate computation of quadrature rules for non-uniform splines. Furthermore, we discuss explicit constructions of optimal quadrature rules for some of the more commonly used uniform splines defined on open knot vectors. First, we introduce basic concepts and define our notation for subsequent use.

### 2.1 Background and notation

A spline is a piecewise polynomial that is characterized by the polynomial degree of its segments and the prescribed regularity at their interfaces. A convenient basis in which to represent polynomial splines is given by B-splines [20, 46].

Let  $\mathcal{P}^p$  denote the space of piecewise polynomials of total degree  $p \geq 0$  and consider the following partitioning of  $[a, b] \subset \mathbb{R}$  into  $n_{el}$  elements,

$$\Omega := \{a = \xi_1 < \xi_2 < \dots < \xi_k < \xi_{k+1} < \dots < \xi_{n_{el}+1} = b\}$$

With every knot  $\xi_k$  we denote an integer,  $\mathbf{r} = \{-1 \leq r_k \leq p - 1, k = 1, \dots, n_{el} + 1\}$ , prescribing the regularity between the polynomial pieces. Then, given the multiplicity at each knot,  $\{p - r_k\}_{k=1}^{n_{el}+1}$ , we can define the *knot vector* as,

$$\mathbf{t} = \{t_i\}_{i=1}^{n+p+1} := \left\{ a = \underbrace{\xi_1, \dots, \xi_1}_{p-r_1}, \dots, \underbrace{\xi_{k-1}, \dots, \xi_{k-1}}_{p-r_{k-1}}, \underbrace{\xi_k, \dots, \xi_k}_{p-r_k}, \dots, \underbrace{\xi_{n_{el}+1}, \dots, \xi_{n_{el}+1}}_{p-r_{n_{el}+1}} = b \right\} \quad (3)$$

With the knot vector in hand, B-splines are stably and efficiently computed using the Cox-DeBoor recursion,

$$N_{i,0}(\xi) = \begin{cases} 1 & \text{if } \xi \in [t_i, t_{i+1}) \\ 0 & \text{otherwise} \end{cases} \quad (4a)$$

$$N_{i,p}(\xi) = \frac{\xi - t_i}{t_{i+p} - t_i} N_{i,p-1}(\xi) + \frac{t_{i+p+1} - \xi}{t_{i+p+1} - t_{i+1}} N_{i+1,p-1}(\xi) \quad (4b)$$

where we define  $0/0 = 0$ .

The space of polynomial splines of degree  $p$  and dimension  $n$ , defined on partition  $\Omega$  with  $r_k$  continuous derivatives at knot  $\xi_k$ , can be written as,

$$\mathcal{S}_{\mathbf{r}}^p(\Omega) = \left\{ \sum_{i=1}^n \alpha_i N_{i,p}(\xi) : \xi \in [a, b], \alpha_i \in \mathbb{R}, i = 1, \dots, n \right\}. \quad (5)$$

In this paper we consider splines with interpolating end-conditions only, which implies that the first and last knot have multiplicity  $p + 1$ , i.e.  $r_1 = r_{n_{el}+1} = -1$ . Such a knot vector is called *open*. Furthermore, many of the spline spaces we shall be interested in are those with uniform internal regularity,  $\{r_k = r, k = 2, 3, \dots, n_{el}\}$ , and defined on a uniform partition, i.e.  $\{\xi_{k+1} - \xi_k = h, k = 1, 2, \dots, n_{el}\}$ . We shall refer to such a spline space as  $\mathcal{S}_r^p(a, b, n_{el})$ .

B-splines have important mathematical properties, many of which are useful in design as well as in analysis. B-spline basis functions of degree  $p$  may have up to  $p - 1$  continuous derivatives, they form a positive partition of unity, and have local support,

$$\text{supp}(N_{i,p}(\xi)) = t_{i+p+1} - t_i. \quad (6)$$

Furthermore, the derivative of a B-spline basis function can be easily computed as,

$$N'_{i,p}(\xi) = p \frac{N_{i,p-1}(\xi)}{t_{i+p} - t_i} - p \frac{N_{i+1,p-1}(\xi)}{t_{i+p+1} - t_{i+1}} \quad (7)$$

The following property is important for our subsequent discussion,

$$\int_{t_i}^{t_{i+p+1}} N_{i,p}(\xi) d\xi = \frac{t_{i+p+1} - t_i}{p + 1}. \quad (8)$$

Non-uniform rational B-splines (NURBS) extend B-splines by the ability to represent shapes with conic sections such as cylinders. A parametrized NURBS curve is written as,

$$\mathbf{C}(\xi) = \sum_{i=1}^n \mathbf{P}_i R_{i,p}(\xi) \quad \text{with} \quad R_{i,p}(\xi) = \frac{w_i N_{i,p}(\xi)}{\sum_{j=1}^n w_j N_{j,p}(\xi)} \quad (9)$$

The coefficients  $\mathbf{P}_i \in \mathbb{R}^d$  are called *control points*. Together they form the so-called *control polygon*. Associated with each  $\mathbf{P}_i$  is a *weight*  $w_i$ . If all weights are set equal to unity the NURBS curve reduces to a B-spline.

Parametric surfaces or volumes can be constructed in a similar fashion. Let  $n_{\text{sd}}$  be the dimension of the spline space and  $\boldsymbol{\xi} = \{\xi^1, \dots, \xi^{n_{\text{sd}}}\}$  denote the parametric coordinates. Given integers representing polynomial degree,  $\mathbf{p} = \{p_1, p_2, \dots, p_{n_{\text{sd}}}\}$ , and corresponding knot vectors,  $\mathbf{t}_1, \dots, \mathbf{t}_{n_{\text{sd}}}$ , a NURBS surface or volume can be constructed as follows,

$$\mathbf{F}(\boldsymbol{\xi}) = \sum_{i=1}^n \mathbf{P}_i R_{i,\mathbf{p}}(\boldsymbol{\xi}) \quad \text{with} \quad R_{i,\mathbf{p}}(\boldsymbol{\xi}) = \frac{w_i N_{i,\mathbf{p}}(\boldsymbol{\xi})}{\sum_{j=1}^n w_j N_{j,\mathbf{p}}(\boldsymbol{\xi})} \quad (10)$$

Here the  $\{\mathbf{P}_i \in \mathbb{R}^d, i = 1, \dots, n\}$  form a rectangular  $n_{\text{sd}}$ -dimensional array of control points,  $i = f(i_1, i_2, \dots, i_{n_{\text{sd}}})$  is a linear index into this array depending on subscripts  $(i_1, i_2, \dots, i_{n_{\text{sd}}})$  and  $N_{i,\mathbf{p}}(\boldsymbol{\xi}) = N_{i_1,p_1}(\xi^1) \cdot N_{i_2,p_2}(\xi^2) \cdot \dots \cdot N_{i_{n_{\text{sd}}},p_{n_{\text{sd}}}}(\xi^{n_{\text{sd}}})$  are the multi-dimensional tensor product B-spline basis functions that are defined by  $\mathbf{p}$  and  $\mathbf{t}_1, \dots, \mathbf{t}_{n_{\text{sd}}}$ .

## 2.2 Global non-linear optimization algorithm for computing optimal quadrature rules for nonuniform splines

In this section we briefly review the global non-linear optimization strategy adopted in [29] for computing optimal quadrature rules for univariate splines. For related work we refer to Karlin [33], in which the conditions for existence and uniqueness of Gaussian rules for arbitrary classes of functions have been introduced, and to [13, 36], where algorithms for their numerical computation have been presented.

Suppose the target space of dimension  $n$  is the spline space  $\mathcal{S}_r^q(\Omega) = \text{span} \{N_{j,q}(\xi), j = 1, 2, \dots, n\}$ . Computing generalized Gaussian quadrature rules for this target space involves finding the zero of the functional  $\mathcal{F} : \mathbb{R}^{2m} \mapsto \mathbb{R}^n$ , defined by,

$$\mathcal{F}_i(\bar{\xi}_1, \dots, \bar{\xi}_m, \bar{w}_1, \dots, \bar{w}_m) = \int_a^b N_{i,q}(\xi) d\xi - \sum_{j=1}^m \bar{w}_j N_{i,q}(\bar{\xi}_j), \quad \text{for } i = 1, 2, \dots, n \quad (11)$$

Since the partial derivatives with respect to the functional are,

$$\frac{\partial \mathcal{F}_i}{\partial \bar{\xi}_k} = \bar{w}_k N'_{i,q}(\bar{\xi}_k) \quad \text{and} \quad \frac{\partial \mathcal{F}_i}{\partial \bar{w}_k} = N_{i,q}(\bar{\xi}_k)$$

the Jacobian matrix of the set of non-linear equations is given by,

$$\nabla \mathcal{F}(\bar{\xi}_1, \dots, \bar{\xi}_m, \bar{w}_1, \dots, \bar{w}_m) = \begin{pmatrix} \bar{w}_1 N'_{1,q}(\bar{\xi}_1) & \cdots & \bar{w}_m N'_{1,q}(\bar{\xi}_m) & N_{1,q}(\bar{\xi}_1) & \cdots & N_{1,q}(\bar{\xi}_m) \\ \vdots & & \vdots & \vdots & & \vdots \\ \bar{w}_1 N'_{n,q}(\bar{\xi}_1) & \cdots & \bar{w}_m N'_{n,q}(\bar{\xi}_m) & N_{n,q}(\bar{\xi}_1) & \cdots & N_{n,q}(\bar{\xi}_m) \end{pmatrix}.$$

If the Jacobian matrix is square, i.e.  $n$  is even and  $2m = n$ , and has a non-vanishing determinant, then the quadrature nodes  $\bar{\xi} = \{\bar{\xi}_j, j = 1, \dots, m\}$  and weights  $\bar{\mathbf{w}} = \{\bar{w}_j, j = 1, \dots, m\}$  are uniquely determined and can be computed using Newton's method, Algorithm 1. If  $n$  is odd then the system may be overdetermined and adjustment is required in order to render the solution unique.

**Remark 2.1.** *A simple solution is proposed in [30]. Consider the target space  $\mathcal{S}_r^q(\Omega)$  of odd dimension  $n$ . Let  $\mathcal{S}_r^q(\tilde{\Omega})$  be obtained by inserting a single knot in the knot vector of  $\mathcal{S}_r^q(\Omega)$ . Then  $\mathcal{S}_r^q(\Omega) \subset \mathcal{S}_r^q(\tilde{\Omega})$ .  $\mathcal{S}_r^q(\tilde{\Omega})$ , which is of even dimension  $n + 1$ , is used as the new target space for quadrature.*

---

### Algorithm 1 Newton method

---

```

1: function NEWTON( $\bar{\xi}_0$ ::Vector{Float} ,  $\bar{\mathbf{w}}_0$ ::Vector{Float})
2:    $\bar{\xi} \leftarrow \bar{\xi}_0$  ▷ Set initial guess quadrature points
3:    $\bar{\mathbf{w}} \leftarrow \bar{\mathbf{w}}_0$  ▷ Set initial guess quadrature weights
4:   while 'Not Converged' do
5:     Solve  $\nabla \mathcal{F}(\bar{\xi}, \bar{\mathbf{w}}) [\Delta \bar{\xi}, \Delta \bar{\mathbf{w}}] = -\mathcal{F}(\bar{\xi}, \bar{\mathbf{w}})$ 
6:      $\bar{\xi} \leftarrow \bar{\xi} + \Delta \bar{\xi}$ 
7:      $\bar{\mathbf{w}} \leftarrow \bar{\mathbf{w}} + \Delta \bar{\mathbf{w}}$ 
8:   end while
9:   return  $\bar{\xi}$  ,  $\bar{\mathbf{w}}$ 
10: end function

```

---

As discussed in [29] and in [30, 14, 41] there are many difficulties that need to be overcome to obtain robust and accurate results. The system to be solved in each Newton step is highly non-linear and can become very ill-conditioned. The latter hampers convergence of

the algorithm towards termination criteria, which could for example be based on the absolute residual,

$$\text{abs}(\mathcal{F}_i(\bar{\xi}_1, \dots, \bar{\xi}_m, \bar{w}_1, \dots, \bar{w}_m)) < \epsilon, \quad i = 1, 2, \dots, n. \quad (12)$$

We propose the following to counteract these difficulties:

1. An improved starting guess.
2. Use of high precision arithmetic such that stringent convergence criteria can be attained.
3. A continuation method, following [36], based on the residual in (11) which makes the Newton method converge to the solution in a series of steps.
4. The reduced singular value decomposition [23, 49] is used to solve the linearized system of equations if they are singular.

Each one of these four ingredients is discussed below. Furthermore, a numerical implementation in the Julia language<sup>1</sup> is provided at <https://github.com/rrhiemstra/quadrules/>.

### 2.2.1 An improved starting guess

The described Newton algorithm will not work unless a reasonably good starting guess of  $\bar{\xi}_0$  and  $\bar{w}_0$  is chosen to initialize it. In [29], Algorithm 1 was initialized with an  $m$ -point Gauss-Legendre rule. In many cases the algorithm failed to converge, especially when the dimension of the target space became large. Recently, an improved starting guess based on an empirical formula was proposed in [30].

One of the contributions of this paper is an improved starting guess. Inspired by [3] we have found that for certain specific target spaces the local structure of the B-spline basis can be exploited to efficiently compute quadrature rules. The approach is best described by an example for an odd degree target spline space.

**Example 2.1.** *Consider a  $C^1$  target space of degree five defined on a uniform partition with an interpolating condition at the left boundary and a periodic condition on the right. Figure 2a depicts the functions whose support ends at the right boundary of the first element. There are four such functions and we can exactly integrate these using just two quadrature points and corresponding weights. This procedure is repeated element by element in Figures 2b to 2d. At every step, four active functions, depicted in solid color, are selected and two quadrature points are computed to match the residual of their integrals. Carefully note that the quadrature points at step  $k$  do not intersect the support of the active functions at step  $k - 1$ . The quadrature rule obtained by this local procedure is therefore exact for all splines in the target space.*

In the case of an even degree  $C^1$  spline space, the procedure needs to be slightly altered. In this case there are an odd number of active functions whose support ends at the right boundary of each element. Steps of two elements need to be taken such that an even number of active functions are selected to initialize the local routine.

---

<sup>1</sup><http://julialang.org>

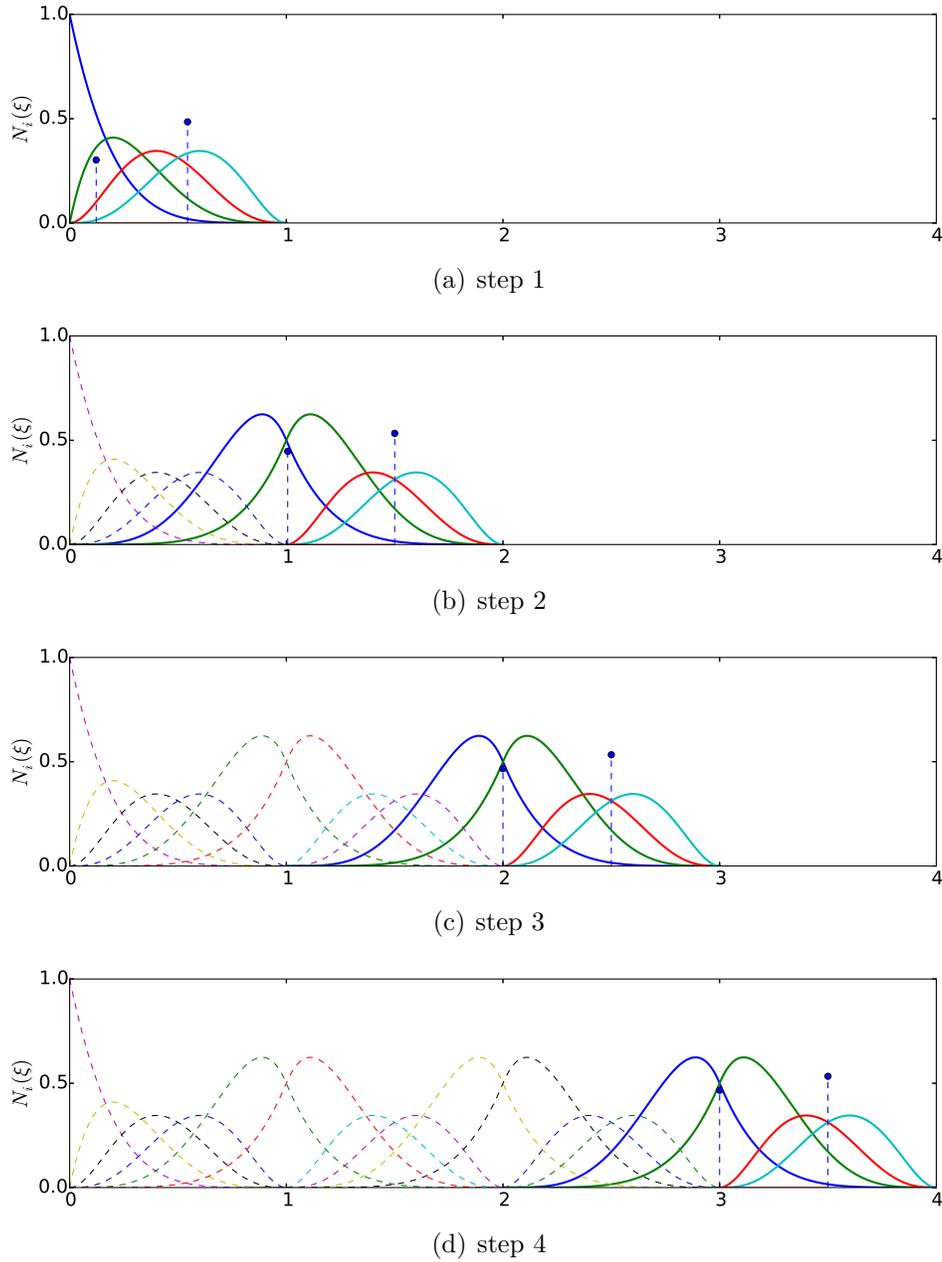


Figure 2: Illustration of the local procedure in the case of odd degree; in this case  $p = 5$ . Note that every step of the algorithm involves one element and four *active* functions which can be exactly integrated with two quadrature points and corresponding weights. The quadrature points at step  $k$ ,  $k = 2, 3, 4$ , do not intersect the support of the active functions at step  $k - 1$ . Therefore the rule is exact for all splines in the target space.

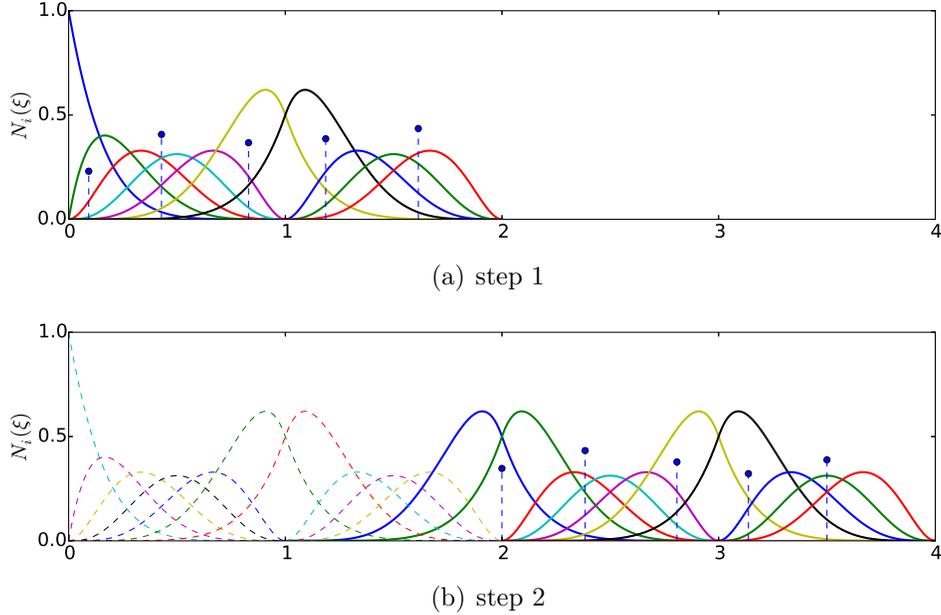


Figure 3: Illustration of the local procedure in the case of even degree; in this case  $p = 6$ . Note that every step of the algorithm involves two elements and ten active functions which can be exactly integrated with five quadrature points and corresponding weights. The quadrature points at step 2 do not intersect the support of the active functions at step 1. Therefore the rule is exact for all splines in the target space.

**Example 2.2.** Consider a  $C^1$  target space of degree six with an interpolating condition at the left boundary and a periodic condition on the right. The ten selected active functions in Figure 3a and Figure 3b can be exactly integrated with a minimum number of five quadrature points and five corresponding weights. The quadrature points at step 2 do not intersect the support of the active functions at step 1. Therefore the rule is exact for all splines in the target space.

Here is a brief summary of the local procedure for computing a quadrature formula:

1. Determine an even number of  $n_{loc} = 2m_{loc}$  active functions whose compact support ends at the right boundary of a local interval  $I_{loc}$ .
2. Use Algorithm 1 initialized with an  $m_{loc}$  point Gauss-Legendre rule to compute the  $m_{loc}$  quadrature points  $\bar{\xi}_0^{loc}$  and  $m_{loc}$  weights  $\bar{\mathbf{w}}_0^{loc}$  in  $I_{loc}$ .
3. Take into account their contribution to the quadrature of the active functions in the following step.

**Remark 2.2.** In most cases of interest, exact rules cannot be obtained using the above described procedure. Nevertheless, an excellent starting guess generally results.

**Example 2.3.** Consider Figure 4 where the approach is outlined for a  $C^1$  spline space of degree five with interpolating end conditions defined on the non-uniform partition  $\{0, 1.2, 2.5, 3.0, 4.2, 5\}$ .

In step three we can observe that the first quadrature point has moved to the left of the knot  $\xi = 2.5$  and therefore intersects with the support of the active functions in step two, preventing their exact quadrature. This also occurs in step five where the first quadrature point prevents the exact quadrature of the active functions of step four. Nevertheless, the obtained quadrature points and weights can still serve as an excellent starting guess.

### 2.2.2 Arbitrary precision arithmetic

The system of equations that need to be solved at each iteration can become highly ill conditioned, leading to a loss of several significant digits in accuracy. In our numerical implementation of Algorithm 1 we use arbitrary precision arithmetic to alleviate this serious problem. Consequently, quadrature rules can be computed to 16 digits of accuracy. Libraries that provide support for arbitrary precision arithmetic are e.g. the GNU Multiple Precision Arithmetic Library<sup>2</sup>, which is available for C and C++. In our work we use the built in support provided by the Julia language.

### 2.2.3 Continuation method

In a continuation method the Newton non-linear solver converges to the solution in a series of steps rather than directly. This greatly improves the robustness in computation when the initial starting guess is not sufficiently accurate. An example is the method of homotopy continuation [8, 9, 30], which transforms known optimal rules from a source space to the target space by deforming the knot vector.

Due to its simplicity, we use a continuation based on the residual of the non-linear equations, see [36]. Instead of using the functional  $\mathcal{F}(\bar{\xi}, \bar{\mathbf{w}})$  in (11) we use,

$$\mathcal{F}^\alpha(\bar{\xi}, \bar{\mathbf{w}}) = \alpha \cdot \mathcal{F}(\bar{\xi}, \bar{\mathbf{w}}) \quad (13)$$

where  $\alpha \in [0, 1]$  is the continuation parameter. For example, if  $\alpha = 1$  we attempt to converge in 1 step to the solution. As in [30], we increase  $\alpha$  from 0 to 1 incrementally using recursive bisection. A vector  $\mathbf{a} = \{a_1, a_2, a_3, \dots, a_k = 1\}$  monitors our progress, and at each iteration we initialize  $\alpha = a_2$ . If a converged solution (success) is obtained, the first element is removed from  $\mathbf{a}$ , reducing its length by one entry. Otherwise we set  $\alpha = \frac{1}{2}(a_2 + a_1)$  and insert  $\alpha$  into the second index of  $\mathbf{a}$ , increasing its length by one entry. Initially we set  $\mathbf{a} = \{0, 1\}$  and the algorithm terminates successfully when  $\dim \mathbf{a} = 1$ .

**Example 2.4.** *Four examples of recursive bisection. At each step  $\alpha$  is denoted by a bar on top.*

1.  $\{0, \bar{1}\} \xrightarrow{\text{success}} \{1, \}$  done
2.  $\{0, \bar{1}\} \xrightarrow{\text{no success}} \left\{0, \frac{\bar{1}}{2}, 1\right\} \xrightarrow{\text{success}} \left\{\frac{1}{2}, \bar{1}\right\} \xrightarrow{\text{success}} \{1, \}$  done
3.  $\{0, \bar{1}\} \xrightarrow{\text{no success}} \left\{0, \frac{\bar{1}}{2}, 1\right\} \xrightarrow{\text{success}} \left\{\frac{1}{2}, \bar{1}\right\} \xrightarrow{\text{no success}} \left\{\frac{1}{2}, \frac{\bar{3}}{4}, 1\right\} \xrightarrow{\text{success}} \left\{\frac{3}{4}, \bar{1}\right\} \xrightarrow{\text{success}} \{1, \}$  done

---

<sup>2</sup><https://gmplib.org/>

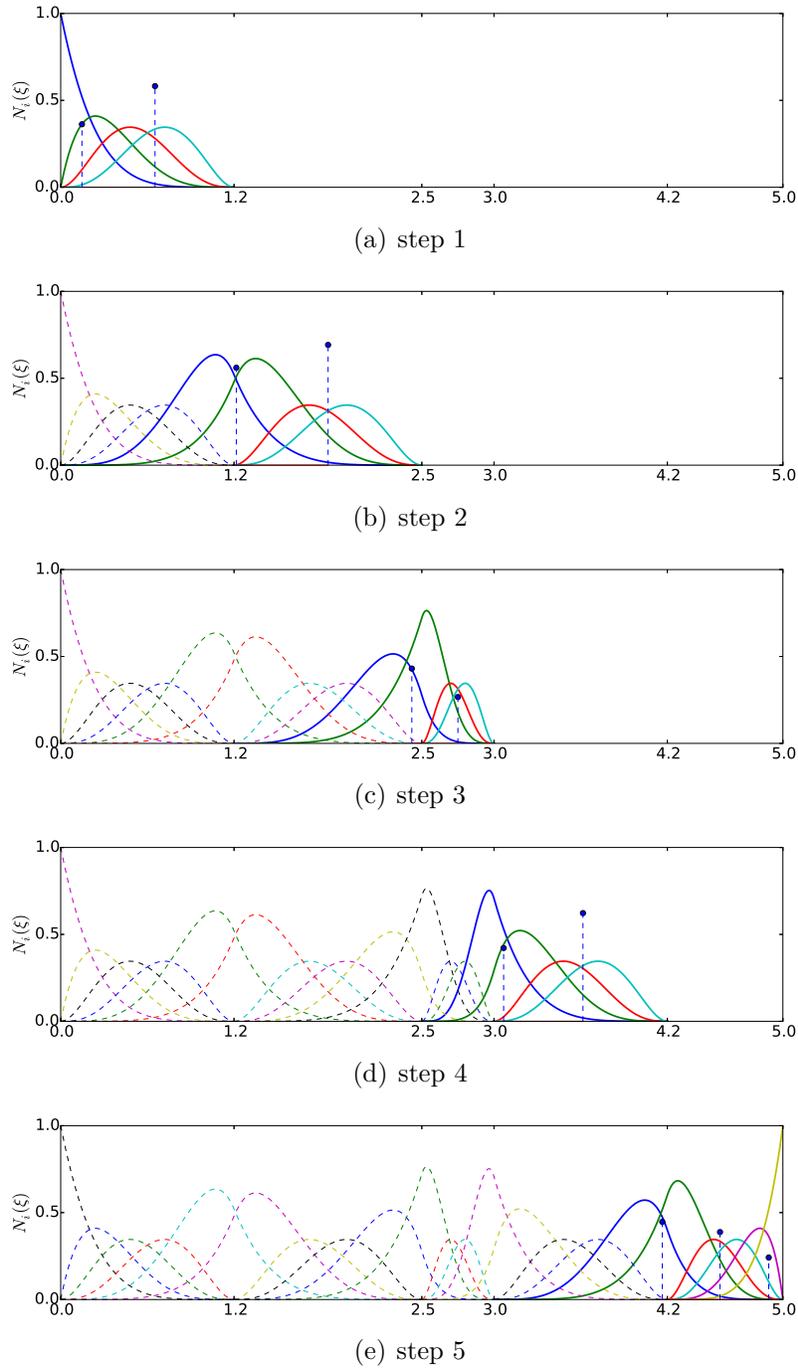


Figure 4: The local approach of computing a quadrature rule applied to the quintic  $C^1$  spline space defined on the non-uniform partition  $\{0, 1.2, 2.5, 3.0, 4.2, 5\}$ . The first quadrature point in step three and step five ends up in the support of the active functions of step two and four, respectively, thereby preventing their exact quadrature. Nevertheless, the resulting points and weights can still serve as an excellent starting guess for initializing Algorithm 1.

$$4. \{0, \bar{1}\} \xrightarrow{\text{no success}} \left\{0, \frac{\bar{1}}{2}, 1\right\} \xrightarrow{\text{no success}} \left\{0, \frac{\bar{1}}{4}, \frac{1}{2}, 1\right\} \xrightarrow{\text{success}} \left\{\frac{1}{4}, \frac{\bar{1}}{2}, 1\right\} \xrightarrow{\text{no success}} \left\{\frac{1}{4}, \frac{\bar{3}}{8}, \frac{1}{2}, 1\right\} \xrightarrow{\text{success}} \left\{\frac{3}{8}, \frac{\bar{1}}{2}, 1\right\} \xrightarrow{\text{success}} \left\{\frac{1}{2}, \bar{1}\right\} \xrightarrow{\text{success}} \{1, \} \text{ done}$$

### 2.2.4 Singular value decomposition

Let  $\mathbf{Ax} = \mathbf{b}$  denote the linearized set of equations that need to be solved in each Newton iteration. The matrix  $A$  can become singular when the layout of the quadrature points and weight is not correct. We have observed that this happens often in the first few Newton iterations when the initial guess is not sufficiently accurate. While the continuation method described increases robustness by changing the right-hand-side,  $\mathbf{b}$ , it will not be of much help if  $A$  is singular. To increase robustness in the first few iterates, we use the reduced singular value decomposition,

$$\mathbf{x} \approx \sum_{i=1}^k \sigma_i^{-1} \mathbf{v}_i \mathbf{u}_i^T \quad (14)$$

where  $k \leq \dim \mathbf{x}$  is chosen such that  $\sigma_k$  is the smallest singular value greater than some positive constant  $C$ , typically  $10^{-5}$ , and  $\mathbf{u}_i$  and  $\mathbf{v}_i$ ,  $i = 1, 2, \dots, k$ , denote the first  $k$  left and right singular vectors of matrix  $A$ , respectively.

## 2.3 Precomputed rules for uniform splines defined by an open knot vector

In this subsection we discuss the assembly of pre-computed quadrature rules for a target space  $\mathcal{S}_r^q(a, b, n_{\text{el}})$ . Some explicit rules of practical interest have been published in [7, 8, 9, 29]. In this paper we introduce the quadrature rule for the space  $\mathcal{S}_1^6(a, b, n_{\text{el}})$ , which is optimal for integrating inner products of  $C^2$  cubic splines and its derivatives, and is consequently of relevance in isogeometric analysis. We further reproduce the rule introduced in [7] for the uniform space  $\mathcal{S}_1^5(a, b, n_{\text{el}})$ , which we propose as an inexpensive alternative to the previous one with slight under-integration.

The dimension  $n$  of a uniform spline space  $\mathcal{S}_r^q(a, b, n_{\text{el}})$  with interpolating end-conditions is easily computed as,

$$n = q + 1 + (n_{\text{el}} - 1)(q - r). \quad (15)$$

and, as discussed, the optimal rule will consist of  $m = \lceil n/2 \rceil$  quadrature points and weights. For sufficiently large  $n_{\text{el}}$  a repetitive structure can be observed away from the boundary [6]. Hence, these rules can be composed of simple building blocks of a boundary rule, a repetitive rule, and as we shall see, a rule local to the center of the domain enforcing symmetry in case  $2m \neq n$ . Because these rules have a special structure, we can precompute their elementary parts and devise an algorithm for their assembly in terms of these simple building blocks.

In Algorithm 2 we outline the main logic in constructing an optimal quadrature rule for splines of degree  $q$  and regularity  $r$  defined by an open knot vector of a uniform mesh of  $n_{\text{el}}$  elements in interval  $(0, n_{\text{el}})$ . The webpage <https://github.com/rrhiemstra/quadrules/>

contains a Matlab implementation of the algorithm we discuss here to generate such quadrature formulae. The repository currently includes rules for the following combinations  $(q, r)$ ,

$$\{(3, 0), (4, 0), (5, 1), (6, 1), (7, 2), (8, 2), (9, 3), (10, 3)\}$$

Rules for even degree spaces are useful for exact integration, whereas the odd degree ones, as we shall see in the sequel of this paper, can be efficiently used in the context of reduced integration.

We discuss the different cases that are covered by the algorithm in the context of several examples related to the space  $\mathcal{S}_1^6(0, n_{\text{el}}, n_{\text{el}})$ . We consider situations in which,

1. The number of elements  $n_{\text{el}}$  is relatively small such that the boundary effect is felt throughout the domain. In lines 13 and 19 of Algorithm 2, the complete rule  $(\bar{\xi}, \bar{w})$  needs to be precomputed. Examples of such rules for the spline space  $\mathcal{S}_1^6(0, n_{\text{el}}, n_{\text{el}})$  for  $n_{\text{el}} = 2, 4, 6, 8, 10$  are given in Table 3.
2. The number of elements  $n_{\text{el}}$  is sufficiently large such that boundary effects and a symmetry effect in case  $2m \neq n$  do not play a role in certain parts of the interval  $(0, n_{\text{el}})$ . In this case we can compose the rule in terms of a boundary rule, a repetitive rule and possibly a center rule enforcing symmetry. There are the following two cases to consider:
  - (a)  $2m = n$ : the rule is unique and automatically symmetric. It is built solely using the boundary rule  $(\bar{\xi}_b, \bar{w}_b)$  and the repetitive rule  $(\bar{\xi}_i, \bar{w}_i)$ . This case is treated in lines 9-11 of Algorithm 2.
  - (b)  $2m = n + 1$  and  $m$  is even: a symmetry condition renders the rule unique. It is built using the boundary rule  $(\bar{\xi}_b, \bar{w}_b)$ , the repetitive rule  $(\bar{\xi}_i, \bar{w}_i)$  and the rule  $(\bar{\xi}_c, \bar{w}_c)$  local to the center of the domain. Lines 15-17 of Algorithm 2 pertain to this case.
  - (c)  $2m = n + 1$  and  $m$  is odd: this case is similar to the case in (b), however, the center rule  $(\bar{\xi}_c, \bar{w}_c)$  is different.

Table 4 illustrates case (a). The first 15 points are contributed to the left boundary. Onwards we can distinguish a repetitive rule up to a certain distance away from the right boundary. The remaining quadrature points and weights are again determined by a boundary rule. Cases (b) and (c) occur when  $2m = n + 1$ . We note that the center rule  $(\bar{\xi}_c, \bar{w}_c)$  depends on  $m$  being even or odd. Selection of the correct data is included in lines 4-8 of Algorithm 2. This leads to two separate types of rules that have different central parts determined by  $(\bar{\xi}_c, \bar{w}_c)$ . Examples are presented in Tables 5 and 6 for  $\mathcal{S}_1^6$ .

**Remark 2.3.** *After computation of a rule on the interval  $(0, n_{\text{el}})$ , the rule can be scaled and translated such that it holds on any interval  $(a, b)$ .*

---

**Algorithm 2** Assembly of quadrature rules using precomputed data: the optimal quadrature rule for the space  $\mathcal{S}_r^q(0, n_{\text{el}}, n_{\text{el}})$  is assembled from the simple building blocks:  $(\bar{\xi}_b, \bar{\mathbf{w}}_b)$ ,  $(\bar{\xi}_i, \bar{\mathbf{w}}_i)$  and  $(\bar{\xi}_c, \bar{\mathbf{w}}_c)$ , referring to a boundary, internal and center rule, respectively.

---

```

1: function QUADRATURERULE( $q::\text{integer}$ ,  $r::\text{integer}$ ,  $n_{\text{el}}::\text{integer}$ )
2:    $n \leftarrow q + 1 + (n_{\text{el}} - 1)(q - r)$  ▷ Dimension of the spline space
3:    $m \leftarrow \lceil \frac{n}{2} \rceil$  ▷ Dimension of the quadrature rule
4:   if  $m$  is even then
5:      $(\bar{\xi}_b, \bar{\mathbf{w}}_b), (\bar{\xi}_i, \bar{\mathbf{w}}_i), (\bar{\xi}_c, \bar{\mathbf{w}}_c) \leftarrow$  load precomputed data from even rule
6:   else if  $m$  is odd then
7:      $(\bar{\xi}_b, \bar{\mathbf{w}}_b), (\bar{\xi}_i, \bar{\mathbf{w}}_i), (\bar{\xi}_c, \bar{\mathbf{w}}_c) \leftarrow$  load precomputed data from odd rule
8:   end if
9:   if  $2m = n$  then
10:    if  $m > 2 \cdot \text{length}(\bar{\xi}_b)$  then
11:       $(\bar{\xi}, \bar{\mathbf{w}}) \leftarrow$  Construct rule from  $(\bar{\xi}_b, \bar{\mathbf{w}}_b)$  and repetitive use of  $(\bar{\xi}_i, \bar{\mathbf{w}}_i)$ .
12:    else
13:       $(\bar{\xi}, \bar{\mathbf{w}}) \leftarrow$  Load  $n_{\text{el}}$ -element precomputed rule
14:    end if
15:  else if  $2m = n + 1$  then
16:    if  $m > 2 \cdot (\text{length}(\bar{\xi}_b) + \text{length}(\bar{\xi}_c))$  then
17:       $(\bar{\xi}, \bar{\mathbf{w}}) \leftarrow$  Construct rule from  $(\bar{\xi}_b, \bar{\mathbf{w}}_b)$ , repetitive use of  $(\bar{\xi}_i, \bar{\mathbf{w}}_i)$  and  $(\bar{\xi}_c, \bar{\mathbf{w}}_c)$ 
18:    else
19:       $(\bar{\xi}, \bar{\mathbf{w}}) \leftarrow$  Load  $n_{\text{el}}$ -element precomputed rule
20:    end if
21:  end if
22:  return  $\bar{\xi}, \bar{\mathbf{w}}$  ▷ Output the quadrature rule
23: end function

```

---

Table 3: Quadrature points (top) and weights (bottom) corresponding to the target spline space  $\mathcal{S}_1^6(0, n_{\text{el}}, n_{\text{el}})$ . Rules are computed for  $n_{\text{el}} = 2, 4, 6, 8$  and 10 elements and are accurate to 16 significant digits.

#	2 elements	4 elements	6 elements	8 elements	10 elements
points					
1	0.0924254744365224	0.0926076787364690	0.0926076787364690	0.0926076787364690	0.0926076787364690
2	0.4275957012000423	0.4284719776081421	0.4284719776081421	0.4284719776081421	0.4284719776081421
3	0.8279244012980120	0.8301893554301429	0.8301893554301429	0.8301893554301429	0.8301893554301429
4	1.1720755987019880	1.1864418084568065	1.1864418084568065	1.1864418084568065	1.1864418084568065
5	1.5724042987999578	1.6139000245489232	1.6139000245489232	1.6139000245489232	1.6139000245489232
6	1.9075745255634775	2.0000000000000000	2.0001087146243037	2.0001087149907884	2.0001087149907884
7	-	2.3860999754510765	2.3869357018258843	2.3869357046428150	2.3869357046428150
8	-	2.8135581915431933	2.8158755443360031	2.8158755522035257	2.8158755522035257
9	-	3.1698106445698571	3.1841244556639969	3.1841245050546592	3.1841245050546592
10	-	3.5715280223918580	3.6130642981741157	3.6130644392673315	3.6130644392673315
11	-	3.9073923212635311	3.9998912853756963	4.0000000000000000	4.000000003658043
12	-	-	4.3860999754510770	4.3869355607326685	4.3869355635486693
13	-	-	4.8135581915431933	4.8158754949453408	4.8158755028125846
14	-	-	5.1698106445698571	5.1841244477964743	5.1841244971874154
15	-	-	5.5715280223918580	5.6130642953571854	5.6130644364513307
16	-	-	5.9073923212635311	5.9998912850092116	5.999999996341957
17	-	-	-	6.3860999754510770	6.3869355607326685
18	-	-	-	6.8135581915431933	6.8158754949453408
19	-	-	-	7.1698106445698571	7.1841244477964743
20	-	-	-	7.5715280223918580	7.6130642953571854
21	-	-	-	7.9073923212635311	7.9998912850092116
22	-	-	-	-	8.3860999754510761
23	-	-	-	-	8.8135581915431942
24	-	-	-	-	9.1698106445698571
25	-	-	-	-	9.5715280223918580
26	-	-	-	-	9.9073923212635311
weights					
1	0.2300483628893541	0.2305048699152140	0.2305048699152140	0.2305048699152140	0.2305048699152140
2	0.4061452268756670	0.4070441617765419	0.4070441617765419	0.4070441617765419	0.4070441617765419
3	0.3638064102349788	0.3671151647471711	0.3671151647471711	0.3671151647471711	0.3671151647471711
4	0.3638064102349788	0.3860513146469310	0.3860513146469310	0.3860513146469310	0.3860513146469310
5	0.4061452268756670	0.4352195321390286	0.4352195321390286	0.4352195321390286	0.4352195321390286
6	0.2300483628893541	0.3481299135502268	0.3484945789563481	0.3484945801852715	0.3484945801852715
7	-	0.4352195321390286	0.4362230043001936	0.4362230076851827	0.4362230076851827
8	-	0.3860513146469310	0.3893473735185717	0.3893473849990721	0.3893473849990721
9	-	0.3671151647471711	0.3893473735185717	0.3893474498446597	0.3893474498446597
10	-	0.4070441617765419	0.4362230043001936	0.4362230993486437	0.4362230993486437
11	-	0.2305048699152140	0.3484945789563481	0.3488588694245676	0.3488588706522378
12	-	-	0.4352195321390286	0.4362230993486437	0.4362231027342958
13	-	-	0.3860513146469310	0.3893474498446597	0.3893474613257502
14	-	-	0.3671151647471711	0.3893473849990721	0.3893474613257502
15	-	-	0.4070441617765419	0.4362230076851827	0.4362231027342958
16	-	-	0.2305048699152140	0.3484945801852715	0.3488588706522378
17	-	-	-	0.4352195321390286	0.4362230993486437
18	-	-	-	0.3860513146469310	0.3893474498446597
19	-	-	-	0.3671151647471711	0.3893473849990721
20	-	-	-	0.4070441617765419	0.4362230076851827
21	-	-	-	0.2305048699152140	0.3484945801852715
22	-	-	-	-	0.4352195321390286
23	-	-	-	-	0.3860513146469310
24	-	-	-	-	0.3671151647471711
25	-	-	-	-	0.4070441617765419
26	-	-	-	-	0.2305048699152140

Table 4: Quadrature points (left) and weights (right) corresponding to the target space  $\mathcal{S}_1^6(0, n_{el}, n_{el})$  in the case that  $2m = n$ . The rule is accurate to 16 significant digits.

	points	weights
1	0.0926076787364690	0.2305048699152140
2	0.4284719776081421	0.4070441617765419
3	0.8301893554301429	0.3671151647471711
4	1.1864418084568065	0.3860513146469310
5	1.6139000245489232	0.4352195321390286
6	2.0001087149907884	0.3484945801852715
7	2.3869357046428150	0.4362230076851827
8	2.8158755522035257	0.3893473849990721
9	3.1841245050546592	0.3893474498446597
10	3.6130644392673315	0.4362230993486437
11	4.0000000003658043	0.3488588706522378
12	4.3869355635486693	0.4362231027342958
13	4.8158755028125846	0.3893474613257502
14	5.1841244971874154	0.3893474613257502
15	5.6130644364513307	0.4362231027342958
1	6+0.0000000000000000	0.3488588718799081
2	6+0.3869355635486691	0.4362231027342958
3	6+0.8158755028125850	0.3893474613257502
4	6+1.1841244971874150	0.3893474613257502
5	6+1.6130644364513309	0.4362231027342958
1	8+0.0000000000000000	0.3488588718799081
2	8+0.3869355635486691	0.4362231027342958
3	8+0.8158755028125850	0.3893474613257502
4	8+1.1841244971874150	0.3893474613257502
5	8+1.6130644364513309	0.4362231027342958
⋮	⋮	⋮

Table 5: Quadrature points (left) and weights (right) corresponding to the target space  $\mathcal{S}_1^6(0, n_{\text{el}}, n_{\text{el}})$  for the case when  $2m = n + 1$  and  $m$  is even.  $c = n_{\text{el}}/2$  represents the symmetry plane. The quadrature points and weights are accurate to 16 significant digits.

	points	weights
1	0.0926076787364690	0.2305048699152140
2	0.4284719776081421	0.4070441617765419
3	0.8301893554301429	0.3671151647471711
4	1.1864418084568065	0.3860513146469310
5	1.6139000245489232	0.4352195321390286
6	2.0001087149907884	0.3484945801852715
7	2.3869357046428150	0.4362230076851827
8	2.8158755522035257	0.3893473849990721
9	3.1841245050546592	0.3893474498446597
10	3.6130644392673315	0.4362230993486437
11	4.0000000003658043	0.3488588706522378
12	4.3869355635486693	0.4362231027342958
13	4.8158755028125846	0.3893474613257502
14	5.1841244971874154	0.3893474613257502
15	5.6130644364513307	0.4362231027342958
<hr/>		
1	6+0.0000000000000000	0.3488588718799081
2	6+0.3869355635486691	0.4362231027342958
3	6+0.8158755028125850	0.3893474613257502
4	6+1.1841244971874150	0.3893474613257502
5	6+1.6130644364513309	0.4362231027342958
<hr/>		
:	:	:
<hr/>		
1	c-6.5000000000000000	0.3488588718799081
2	c-6.1130644364513307	0.4362231027342958
3	c-5.6841244971874154	0.3893474613257502
4	c-5.3158755028125846	0.3893474613257502
5	c-4.8869355635486693	0.4362231027342958
6	c-4.50000000000005276	0.3488588718781366
7	c-4.1130644364553941	0.4362231027294106
8	c-3.6841244971987668	0.3893474613091839
9	c-3.3158755028838502	0.3893474612156242
10	c-2.8869355637522220	0.4362231025972326
11	c-2.5000041259055168	0.3488450258123363
12	c-2.1130961962162917	0.4361849197447470
13	c-1.6842132016006659	0.3892181373293682
14	c-1.3164333038404137	0.3884825476394018
15	c-0.8885488201152921	0.4350935951843146
16	c-0.5127080451864184	0.3136422475372353
17	c-0.1889822365046136	0.3629629629629630

Table 6: Quadrature points (left) and weights (right) corresponding to the target space  $\mathcal{S}_1^6(0, n_{\text{el}}, n_{\text{el}})$  for the case when  $2m = n + 1$  and  $m$  is odd.  $c = n_{\text{el}}/2$  represents the symmetry plane. The quadrature points and weights are accurate to 16 significant digits.

	points	weights
1	0.0926076787364690	0.2305048699152140
2	0.4284719776081421	0.4070441617765419
3	0.8301893554301429	0.3671151647471711
4	1.1864418084568065	0.3860513146469310
5	1.6139000245489232	0.4352195321390286
6	2.0001087149907884	0.3484945801852715
7	2.3869357046428150	0.4362230076851827
8	2.8158755522035257	0.3893473849990721
9	3.1841245050546592	0.3893474498446597
10	3.6130644392673315	0.4362230993486437
11	4.0000000003658043	0.3488588706522378
12	4.3869355635486693	0.4362231027342958
13	4.8158755028125846	0.3893474613257502
14	5.1841244971874154	0.3893474613257502
15	5.6130644364513307	0.4362231027342958
<hr/>		
1	6+0.0000000000000000	0.3488588718799081
2	6+0.3869355635486691	0.4362231027342958
3	6+0.8158755028125850	0.3893474613257502
4	6+1.1841244971874150	0.3893474613257502
5	6+1.6130644364513309	0.4362231027342958
<hr/>		
⋮	⋮	⋮
<hr/>		
1	c-5.1130644364513307	0.4362231027342958
2	c-4.6841244971874154	0.3893474613257502
3	c-4.3158755028125855	0.3893474613257500
4	c-3.8869355635486693	0.4362231027342956
5	c-3.5000000049975220	0.3488588551078113
6	c-3.1130644749227625	0.4362230564804395
7	c-2.6841246046675309	0.3893473044744377
8	c-2.3158761776286205	0.3893464183818660
9	c-1.8869374920404343	0.4362218019613904
10	c-1.5004028451009870	0.3475146459826823
11	c-1.1161473022682864	0.4325315154620461
12	c-0.6925086935571694	0.3781951324919341
13	c-0.3624724648874144	0.3220111278283742
14	c-0.0000000000000000	0.3883591555379455

### 3 Application of optimal quadrature rules in isogeometric analysis

Numerical quadrature is a key ingredient in any isoparametric finite element technology, including isogeometric analysis. In this section we discuss quadrature rules that are of practical interest in isogeometric analysis based on tensor product NURBS and hierarchical B-splines. Quadrature rules that take inter-element smoothness of isogeometric function spaces into consideration are considerably more efficient than the commonly used Gauss-Legendre rules. The new rules can also be used effectively for reduced integration, another mechanism through which additional efficiency can be obtained without affecting accuracy.

#### 3.1 Problem statement

Let  $n_{\text{sd}} (= 2 \text{ or } 3)$  denote the number of space dimensions and let  $\Omega \subset \mathbb{R}^{n_{\text{sd}}}$  be an open set with piecewise smooth boundary  $\Gamma = \overline{\Gamma_g \cup \Gamma_h}$  with unit outward normal vector  $\mathbf{n}$ . Then given a scalar source field  $f$  and positive constants  $\kappa$  and  $\alpha$  we wish to determine the scalar field  $u$  such that,

$$-\nabla \cdot (\kappa \nabla u) + \alpha u = f \quad \text{in } \Omega \tag{16a}$$

$$u = g \quad \text{on } \Gamma_g \tag{16b}$$

$$\kappa \nabla u \cdot \mathbf{n} = h \quad \text{on } \Gamma_h \tag{16c}$$

We wish to solve (16) using Galerkin's method. The cost of evaluating boundary integrals, that enter the weak formulation after multiplying (16a) by a test function and integrating by parts, is negligible as compared to numerical computation of the domain integrals. This is independent of the specific quadrature rule used. Hence, to simplify matters we shall in the following assume homogeneous Dirichlet and Neumann data, that is  $g = h = 0$ . The equivalent weak formulation now reads: find  $u \in V$  such that for all  $v \in V$ ,

$$a(u, v) = l(v)$$

where,

$$a(u, v) = \int_{\Omega} (\nabla v \cdot \kappa \nabla u + v \alpha u) d\Omega, \tag{17a}$$

$$l(v) = \int_{\Omega} v f d\Omega. \tag{17b}$$

With the chosen boundary conditions, the test and trial space are equivalent. The function space  $V$  is typically thought of as  $H_0^1(\Omega)$ , the space of square integrable functions on  $\Omega$  with square integrable derivatives and zero Dirichlet boundary conditions.

We assume that  $\Omega$  can be discretized in terms of a well defined NURBS parametrization  $\mathbf{F} : \Omega' \rightarrow \Omega$  with coordinates  $\boldsymbol{\xi} = (\xi_1, \xi_2, \dots, \xi_{n_{\text{sd}}})$ , see Section 2.1. In isogeometric analysis, we then proceed according to the isoparametric finite element paradigm and substitute the

trial function  $u \in V$  and test functions  $v \in V$  by finite dimensional spline representations  $u^h, v^h \in V^h$  that are discretized by means of the same NURBS basis functions,

$$V^h := \text{span} \{ \mathbf{R}_i \circ \mathbf{F}^{-1}, i = 1, 2, \dots, n \} \subset V$$

The objective is to find the finite dimensional approximation  $u^h \in V^h$  such that for all  $v^h \in V^h$ ,

$$a(u^h, v^h) = l(v^h) \quad (18)$$

Relying on numerical integration to compute the domain integrals present in the bilinear and linear forms  $a(\cdot, \cdot)$  and  $l(\cdot)$ , we actually solve the closely related problem: find  $\tilde{u}^h \in V^h$  such that for all  $v^h \in V^h$ ,

$$a^*(\tilde{u}^h, v^h) = l^*(v^h) \quad (19)$$

where,

$$a^*(u^h, v^h) = \sum_{k=1}^m [\nabla v^h(\mathbf{x}_k) \cdot \kappa \nabla u^h(\mathbf{x}_k) + v^h(\mathbf{x}_k) \alpha u^h(\mathbf{x}_k)] \bar{w}_k \quad (20a)$$

$$l^*(v^h) = \sum_{k=1}^m v^h(\mathbf{x}_k) f(\mathbf{x}_k) \cdot \bar{w}_k \quad (20b)$$

Here  $\{\mathbf{x}_k = \mathbf{F}(\bar{\boldsymbol{\xi}}_k) \in \Omega, k = 1, 2, \dots, m\}$  is a set of quadrature points that are images of those given in the parametric domain and  $\{\bar{w}_k \in \mathbb{R}^+, k = 1, 2, \dots, m\}$  is the set of corresponding quadrature weights.

**Remark 3.1.** *In this work we focus solely on quadrature rules that obey a tensor product structure. As such these rules can be applied in isogeometric analysis based on NURBS and possibly in a multilevel setting based on local (truncated) hierarchical refinement [12, 22, 34, 42, 43, 50]. Although they are defined in a patch-wise manner, application of the proposed rules can be done in an element-wise fashion, since quadrature points that are inside a specific Bézier element can easily be identified. Hence, there is little difference in applying these rules as compared with Gauss-Legendre quadrature.*

## 3.2 Choice of quadrature

Usually the goal of the quadrature rule is to integrate all products such that the difference,

$$a(u^h, v^h) - a^*(u^h, v^h) = r \quad \text{with } u^h, v^h \in V^h, \quad (21)$$

is “small”. In practice this is typically done as follows. We assume the target space is composed of piecewise polynomials and the geometric mapping is affine. Then we ask that the quadrature rule exactly integrates all functions present in the bilinear form  $a(\cdot, \cdot)$ . We shall refer to this case as *full integration* or *full quadrature*.

However, it is a misconception that  $a(u^h, v^h)$  and  $a^*(u^h, v^h)$  need to be “close” in order for  $u^h$  and  $\tilde{u}^h$  to be “close”. In Section 5 we determine the order of accuracy of the quadrature

rule such that the error due to numerical integration,  $u^h - \tilde{u}^h$ , is of the same order to that of the discretization error,  $u - u^h$ . In this case the target space is chosen smaller than required to exactly evaluate the integrands in the bilinear form  $a(\cdot, \cdot)$ , yet full accuracy, that is, optimal convergence rates, may still be attained. Opposed to full integration we call this *reduced integration* or *reduced quadrature*.

Let the trial and test space be given by the spline spaces  $V^h := \mathcal{S}_{\mathbf{r}_1}^{p_1} \otimes \mathcal{S}_{\mathbf{r}_2}^{p_2} \otimes \dots \otimes \mathcal{S}_{\mathbf{r}_{n_{sd}}}^{p_{n_{sd}}}$  with the correct boundary conditions built in. Then the integrand of the bilinear form  $a(\cdot, \cdot)$  under consideration is an element of the following space,

$$\nabla v^h \cdot \kappa \nabla u^h + v \alpha u \in \mathcal{S}_{\mathbf{r}_1-1}^{2p_1} \otimes \mathcal{S}_{\mathbf{r}_2-1}^{2p_2} \otimes \dots \otimes \mathcal{S}_{\mathbf{r}_{n_{sd}}-1}^{2p_{n_{sd}}} \quad (22)$$

In this paper we shall consider the following paradigms of full and reduced integration,

1. Full integration corresponding to the target space  $\mathcal{S}_{\mathbf{r}_1-1}^{2p_1} \otimes \mathcal{S}_{\mathbf{r}_1-1}^{2p_1} \otimes \dots \otimes \mathcal{S}_{\mathbf{r}_{n_{sd}}-1}^{2p_{n_{sd}}}$ .
2. Reduced integration corresponding to a target space  $\mathcal{S}_{\mathbf{r}_1-1}^{2p_1-1} \otimes \mathcal{S}_{\mathbf{r}_1-1}^{2p_1-1} \otimes \dots \otimes \mathcal{S}_{\mathbf{r}_{n_{sd}}-1}^{2p_{n_{sd}}-1}$ .

Considering a trial and test space consisting of bi-quadratic splines  $\mathcal{S}_1^2 \otimes \mathcal{S}_1^2$ , the integrand in  $a^*(\cdot, \cdot)$  will be in the space  $\mathcal{S}_0^4 \otimes \mathcal{S}_0^4$ . At the element level this integrand will involve all monomials depicted in Figure 5. In full quadrature we choose the rule that integrates all these monomials with the least number of quadrature points. This is the rule corresponding to the target space  $\mathcal{S}_0^4 \otimes \mathcal{S}_0^4$ . In reduced quadrature we disregard some of the products in  $a^*(\cdot, \cdot)$  and choose the quadrature rule corresponding to the target space  $\mathcal{S}_0^3 \otimes \mathcal{S}_0^3$ . The monomials that are exactly integrated on each Bézier element are those denoted in dark gray in Figure 5.

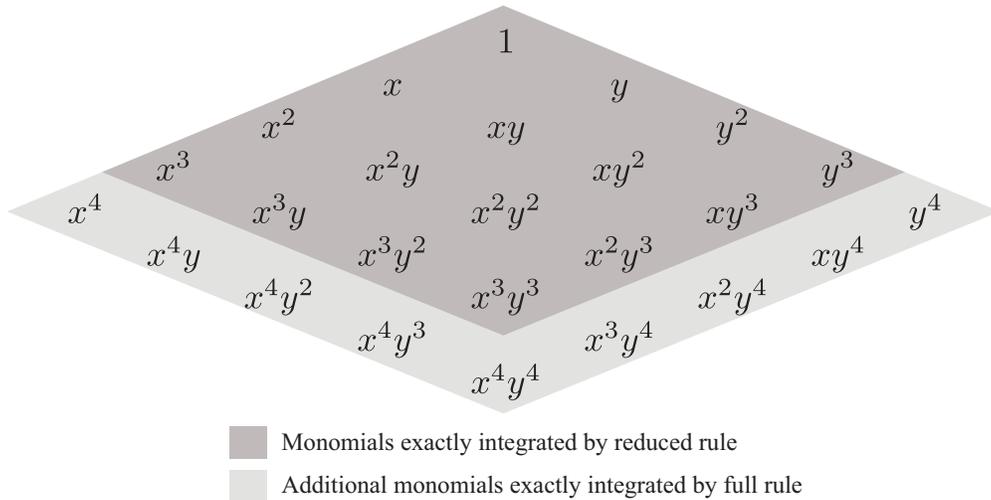


Figure 5: Monomials that are exactly integrated by reduced and full quadrature.

Full quadrature automatically leads to a symmetric positive definite set of matrix equations, with a unique solution  $\tilde{u}^h$ , that is, under some mild assumptions, as “close” as possible to  $u^h$ . In Section 5 we prove that our reduced rules are stable and are accurate in the  $H^1$ -norm.

### 3.3 Numerical examples: application to NURBS

In the following two subsections we compare full and reduced integration using the optimal rules with Gauss-Legendre integration in the context of two- and three-dimensional NURBS based isogeometric analysis.

#### 3.3.1 Quarter annulus: 2d problem with a smooth solution

To test the full and reduced quadrature rules in two dimensional NURBS based isogeometric analysis, we use the model problem (16) with  $\alpha = 1$  and its Galerkin variational formulation (18) defined over a quarter of an annular section that is discretized by a single patch of quadratic NURBS. Geometry and boundary conditions, the initial  $9 \times 6$  mesh, and the corresponding solution are illustrated in Figure 6. The quarter annulus is located within the positive quadrant of the Cartesian coordinate system  $(x, y)$ . The source term,

$$f = 16r^2 \sin(x) - 68 \sin(x) + x(8r^2 - 68) \cos(x) \quad (23)$$

is manufactured in such a way that the exact solution to (1) over the quarter annulus is

$$u = (r^2 - 1)(r^2 - 16) \sin(x) \quad (24)$$

where  $r = \sqrt{x^2 + y^2}$  denotes the radial coordinate.

Figure 7 illustrates the full and reduced quadrature rules for the coarsest mesh corresponding to quadratic and cubic NURBS discretization. Focussing on the quadratic case: Figure 7a (left) depicts the quadrature points of the full Gauss-Legendre rule that requires 9 points per element, and will serve as the reference. The middle and right figures show the points corresponding to full and reduced quadrature, which require asymptotically 4 and slightly more than 2 points per Bézier element, respectively. In the cubic case, Figure 7b, reduced and full quadrature rules compare favorably with 4 and about 6 points per Bézier element as compared to Gauss-Legendre quadrature with 16 points. These examples clearly illustrate the possible savings in quadrature effort, which scales nearly linearly with element formation time.

In Figure 8 the accuracy is compared between the different quadrature rules under uniform mesh refinement as a function of the number of elements. The accuracy is measured for quadratic, cubic, and quartic discretizations, in terms of the relative error in the  $L^2$ -norm and the  $H^1$ -semi-norm. The results confirm that the reduced quadrature rules preserve optimal rates of convergence, under geometric transformations, for polynomial degrees 2, 3, and 4. In fact, hardly any degradation in accuracy is observed, as compared with full quadrature of the optimal and Gauss-Legendre rule. Furthermore, no problems with regard to stability were observed in application of the reduced rules.

#### 3.3.2 Cylindrical section: 3d problem with a smooth solution

To test the optimal full and reduced rules for three-dimensional NURBS based isogeometric analysis, we use the model problem (16) with  $\alpha = 0$  and its Galerkin variational formulation (18) defined over a quarter of a cylindrical section that is discretized by a single patch of

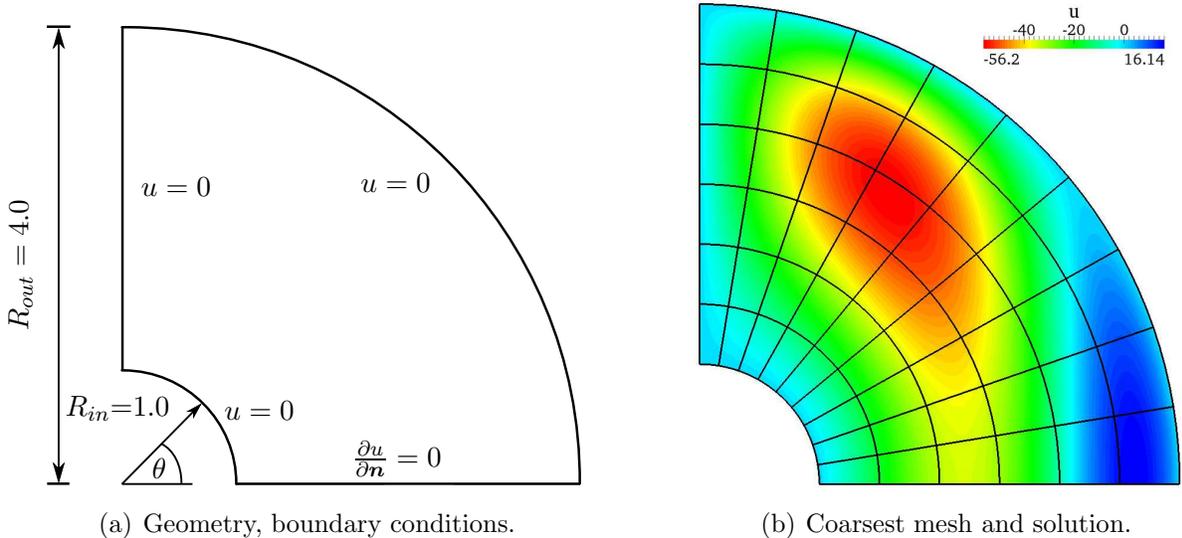


Figure 6: Source problem on a 2D annular domain discretized with one patch of quadratic NURBS.

quadratic NURBS basis functions. The quarter is located within the positive octant of the Cartesian coordinate system  $(x, y, z)$ . The source term,

$$f = \frac{(\pi r(4L^2 + 1) \sin(\pi(2r - 1)) - 2L^2 \cos(\pi(2r - 1))) \cdot \pi \sin(\pi z/L)}{L^2 r} \quad (25)$$

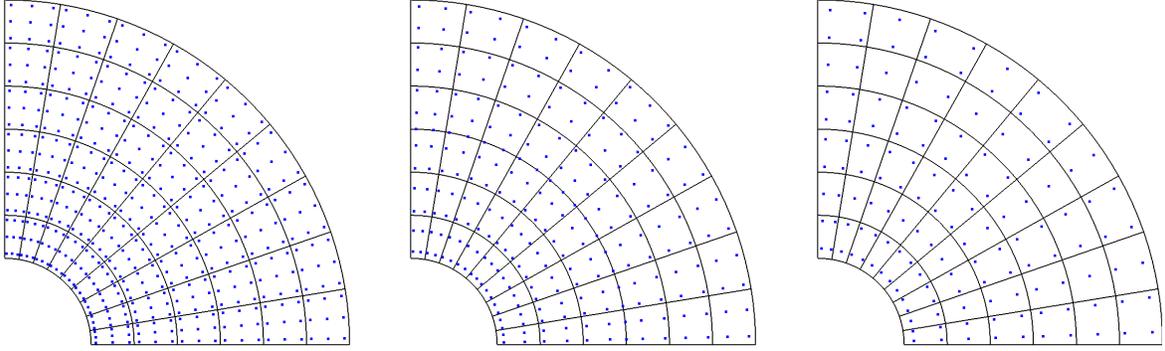
is manufactured in such a way that the exact solution to the model problem over the quarter cylindrical section is,

$$u = \sin\left(\frac{\pi z}{L}\right) \sin(\pi(r - 1)) \quad (26)$$

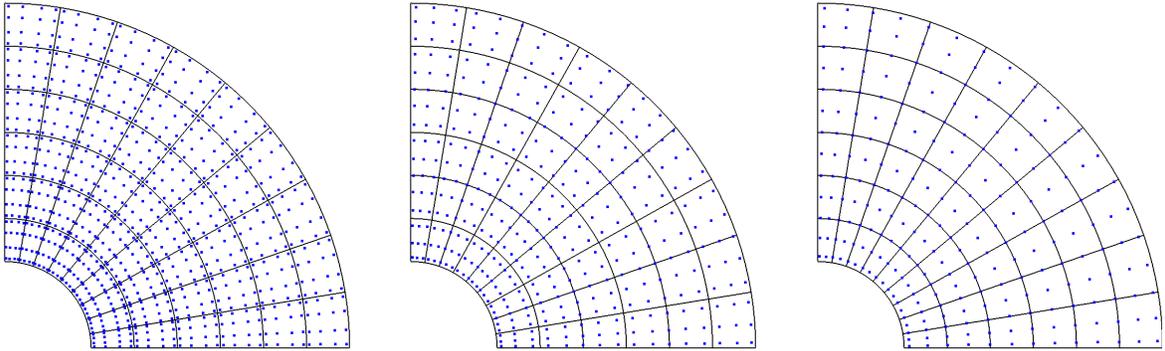
where  $r = \sqrt{x^2 + y^2}$  denotes the radial coordinate. Geometry, boundary conditions, the initial  $3 \times 6 \times 5$  mesh, and the corresponding solution are illustrated in Figure 9.

Figures 10a and b depict the distribution of the quadrature points for quadratic and cubic discretization on a coarse mesh. Significant savings in quadrature points can be observed for the optimal full and reduced quadrature rules as compared to the Gauss-Legendre rule. For both the quadratic and the cubic discretization the optimal full quadrature is roughly four times as efficient in the asymptotic limit. The savings for the reduced rule are about a factor of eight, as compared with the Gauss-Legendre rule.

Regarding accuracy, we report similar results as for the two-dimensional smooth problem. In Figure 11 we compare the accuracy obtained with the different quadrature rules under uniform mesh refinement as a function of the number of elements. The accuracy is measured for quadratic, cubic, and quartic discretizations, all in terms of the relative error in the  $L^2$ -norm and the  $H^1$ -semi-norm. Again optimal rates of convergence are observed for the reduced quadrature rule, under geometric transformations, for all polynomial degrees. In fact, hardly any degradation in accuracy is observed, as compared with full quadrature of the optimal and Gauss-Legendre rule. No problems with regard to stability have been observed with the reduced rules, just as predicted by the theory discussed in Section 5.



(a) Quadratic discretization: (left) Full Gauss quadrature with 9.00 points, (middle) optimal quadrature with 4.00 points, and (right) reduced quadrature with 2.25 points per Bezier element.



(b) Cubic discretization: (left) Full Gauss quadrature with 16.00 points, (middle) optimal quadrature with 6.25 points, and (right) reduced quadrature with 4.00 points per Bezier element.

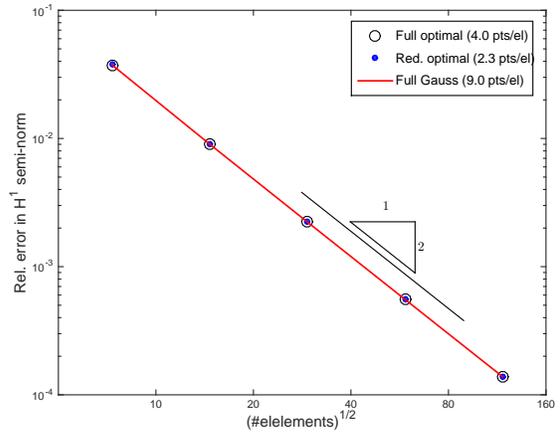
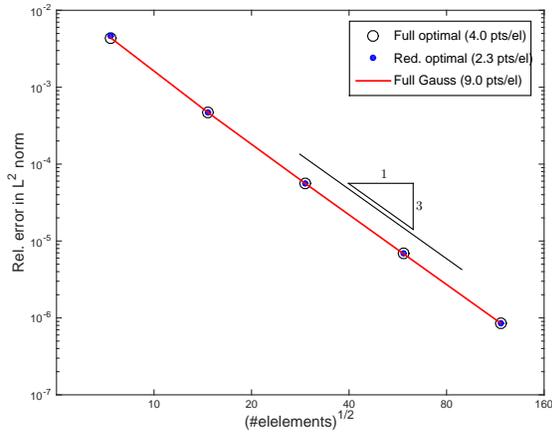
Figure 7: Distribution of the quadrature points for the 2D model problem on the coarse mesh.

### 3.4 Numerical examples: application to multilevel hierarchical splines

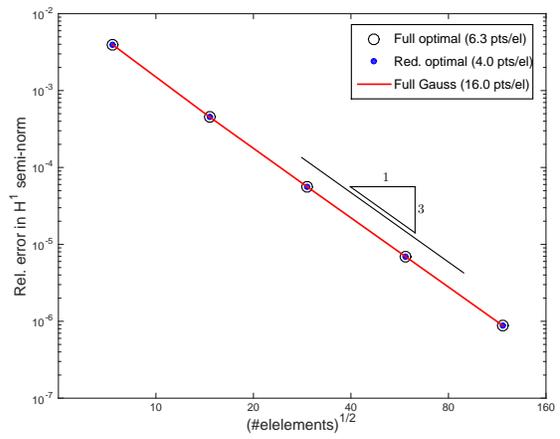
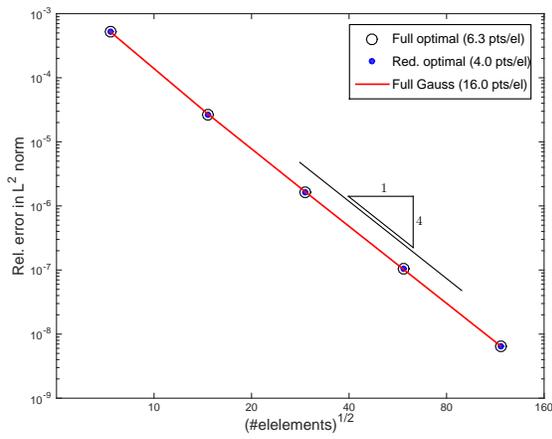
Multilevel hierarchical B-splines [34] provide local refinement through a hierarchy of nested spline spaces. More recent work includes its application in isogeometric analysis [12, 42, 50] and its extension to the so-called truncated bases [22]. Because they are still based on the tensor product structure of the underlying spline spaces we can apply the proposed patch-wise rules to isogeometric analysis based on hierarchical B-splines. After a brief overview of hierarchical B-splines we test the efficacy of the optimal full quadrature rules applied to two- and three-dimensional problems for which the exact solution has a corner singularity.

#### 3.4.1 Multilevel hierarchical B-splines

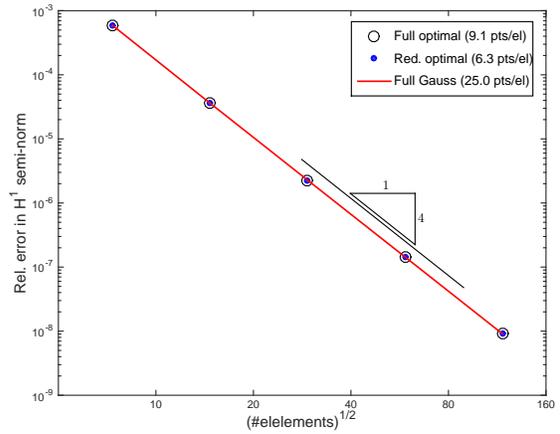
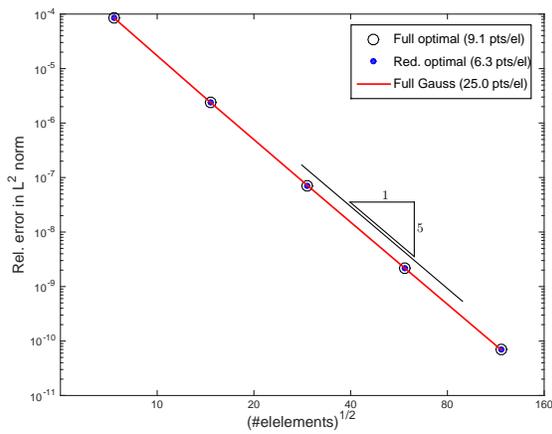
We consider the univariate setting. Extension to the multivariate case is performed analogously. In a multilevel discretization  $\mathcal{MS}(\Omega', n_l)$  based on hierarchical B-splines we consider



(a) Quadratic discretization:  $L^2$ -norm (left) and  $H^1$  semi-norm (right)



(b) Cubic discretization:  $L^2$ -norm (left) and  $H^1$  semi-norm (right)



(c) Quartic discretization:  $L^2$ -norm (left) and  $H^1$  semi-norm (right)

Figure 8: Convergence under global h-refinement for the 2D reaction-diffusion problem. The square root of the total number of elements is taken as a measure of the characteristic mesh size.

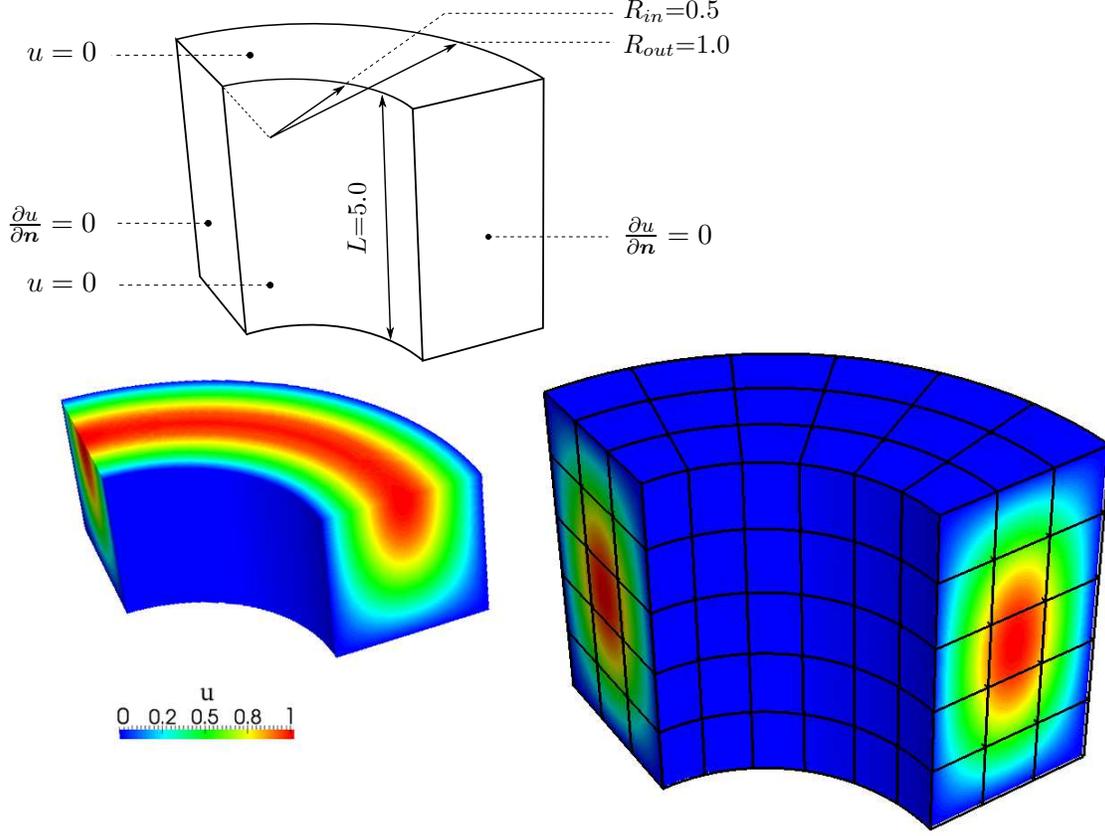


Figure 9: Model problem defined on a 3D cylindrical section discretized with one patch of quadratic NURBS basis functions.

a sequence of  $n_l$  nested spline spaces,

$$\mathcal{S}_p^{(1)}(\Omega', n_{\text{el}}) \subset \mathcal{S}_p^{(2)}(\Omega', 2n_{\text{el}}) \subset \mathcal{S}_p^{(3)}(\Omega', 4n_{\text{el}}) \subset \dots \subset \mathcal{S}_p^{(n_l)}(\Omega', 2^{n_l-1} \cdot n_{\text{el}})$$

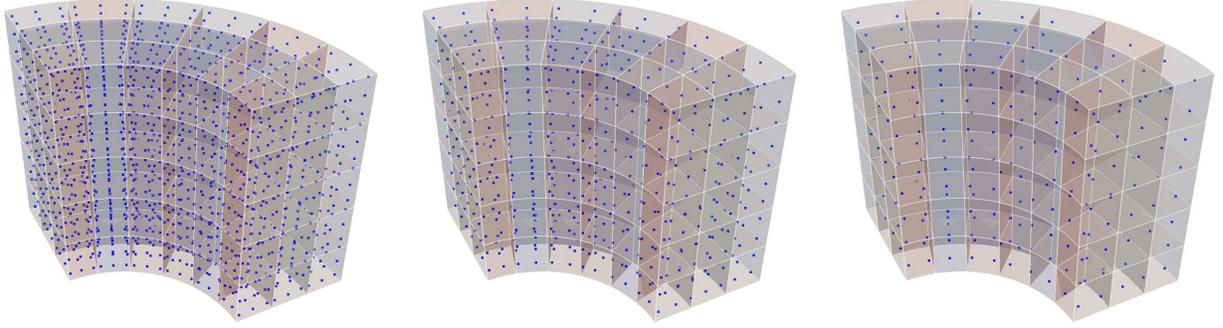
Because each basis function on level  $k$ ,  $N_{i,p}^{(k)}(\xi) \in \mathcal{S}_p^{(k)}(\Omega', 2^{k-1} \cdot n_{\text{el}})$ , can also be represented in the refined space,  $\mathcal{S}_p^{(k+1)}(\Omega', 2^k \cdot n_{\text{el}})$ , we can construct a selection scheme that locally deactivates basis functions on level  $k$  and replaces them with functions in level  $k+1$ . This increases local mesh resolution without refinement propagating throughout  $\Omega'$ .

A hierarchical spline space can be defined as follows,

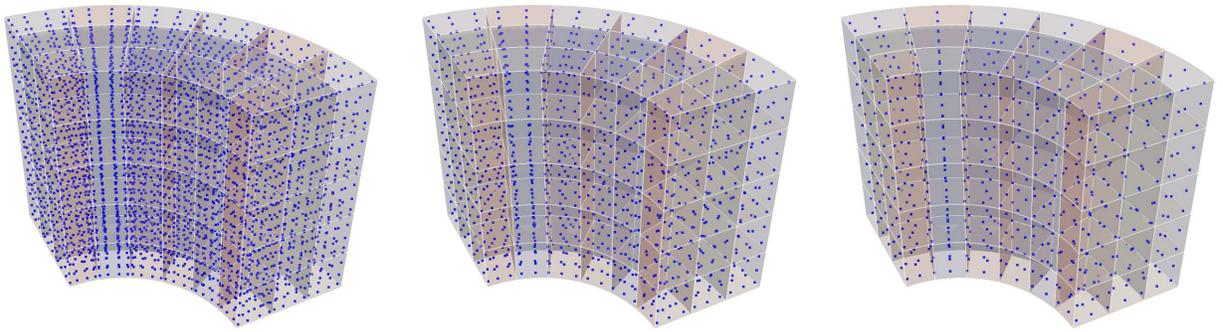
$$\mathcal{MS}(\Omega', n_l) = \text{span} \left\{ N_{i,p}^{(k)}(\xi), \text{ for } i \in I^{(k)}, k = 1, 2, \dots, n_l \right\} \quad (27)$$

Here  $N_{i,p}^{(k)}(\xi)$ ,  $i \in I^{(k)}$  are B-spline basis functions active on level  $k$  and  $I^{(k)}$ ,  $k = 1, 2, \dots, n_l$  are the sets of indices that track the basis functions that are active in the multilevel spline space. This is constructed such that the space spans the full set of polynomials and global linear independence is maintained.

Selection of the correct quadrature rule is then straightforward. Given that entries of the bilinear form  $a^*(N_{i,p}^{(k)}, N_{i,p}^{(l)})$  in Section 3.1 need to be evaluated, we can consider the two following cases:



(a) Quadratic discretization: (left) Full Gauss quadrature with 27.00 points, (middle) optimal quadrature with 8.00 points, and (right) reduced quadrature with 3.38 points per Bezier element.



(b) Cubic discretization: (left) Full Gauss quadrature with 64.00 points, (middle) optimal quadrature with 15.60 points, and (right) reduced quadrature with 8.00 points per Bezier element.

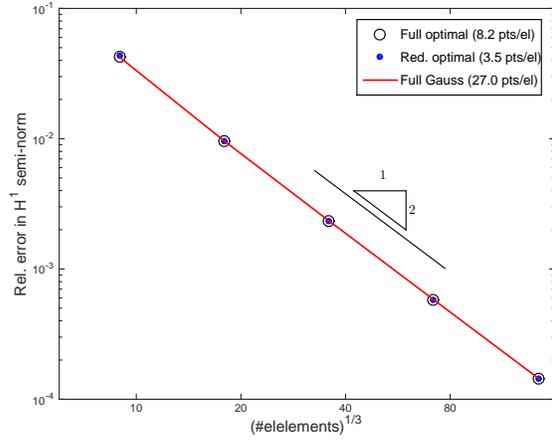
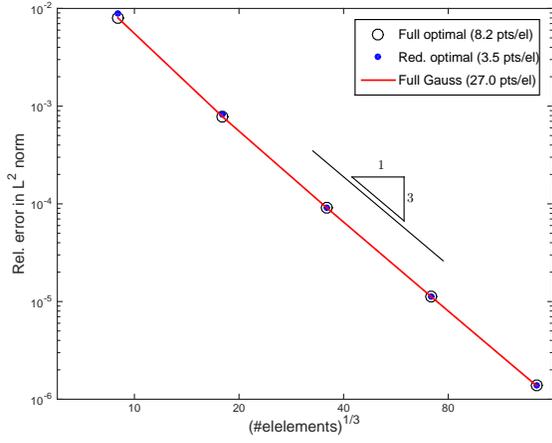
Figure 10: Distribution of the quadrature points for the 3D model problem on the coarse mesh.

1. If  $k \geq l$  we use the rule corresponding to the space, i.e.  $\mathcal{S}_{2p,r-1}^{(k)}(\Omega, 2^{k-1} \cdot n_{el})$ .
2. If  $k < l$  we use the rule corresponding to the space, i.e.  $\mathcal{S}_{2p,r-1}^{(l)}(\Omega, 2^{l-1} \cdot n_{el})$ .

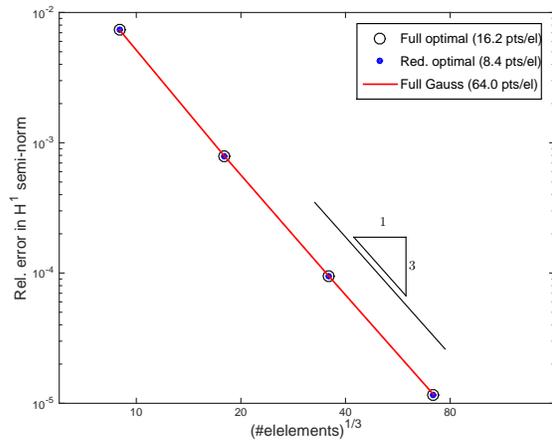
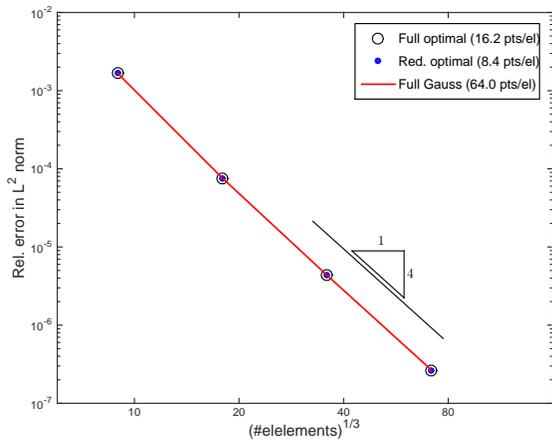
**Remark 3.2.** *Note that selection of the quadrature rule and subsequent integration of the entries of the stiffness and mass matrices is done function-wise. A formation routine that reflects this structure would therefore be the natural choice.*

### 3.4.2 L-shaped domain: 2d problem with a rough solution, multi-patch and adaptive local refinement

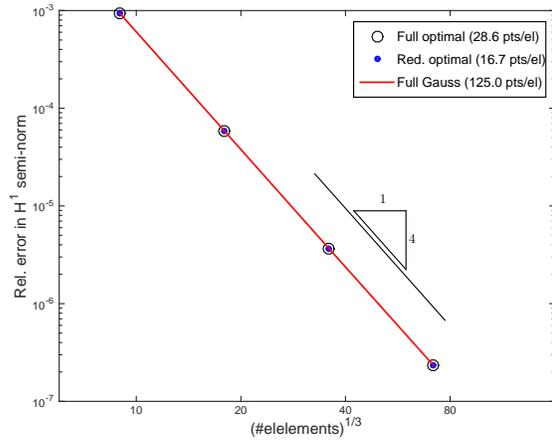
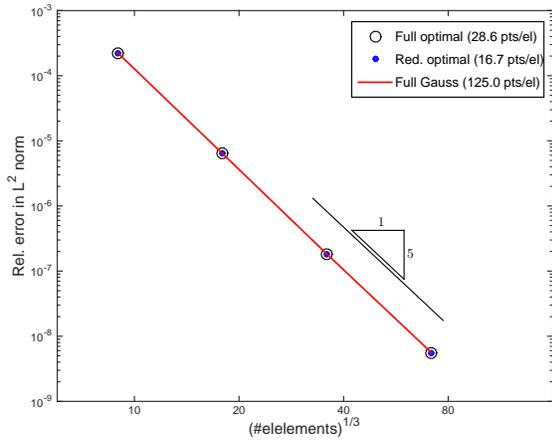
The next numerical example consists of a stationary heat conduction problem,  $\alpha = 0$ , defined over an L-shaped domain,  $\Omega = [-1, 1]^2 \setminus ([0, 1] \times [-1, 0])$ . Homogeneous Dirichlet boundary conditions are set on the reentrant edges  $\Gamma_{D,1} = [0 \leq x \leq 1]$  and  $\Gamma_{D,2} = [-1 \leq y \leq 0]$ , and on the remaining boundaries, Neumann conditions are imposed that satisfy the exact



(a) Quadratic discretization:  $L^2$ -norm (left) and  $H^1$  semi-norm (right)



(b) Cubic discretization:  $L^2$ -norm (left) and  $H^1$  semi-norm (right)



(c) Quartic discretization:  $L^2$ -norm (left) and  $H^1$  semi-norm (right)

Figure 11: Convergence under global h-refinement for the 3D diffusion problem. The cube root of the total number of elements is taken as a measure of the characteristic mesh size.

solution,

$$T = r^{\frac{2}{3}} \sin\left(\frac{2\phi}{3}\right), \quad (28)$$

with polar coordinates  $r = \sqrt{x^2 + y^2}$  and  $\phi = \arccos(x/r)$ . The solution (28) can be characterized as “rough” since its gradient exhibits a singularity at the reentrant corner. The exact solution and its first derivative with respect to  $r$  are plotted in Figures 12a and b.

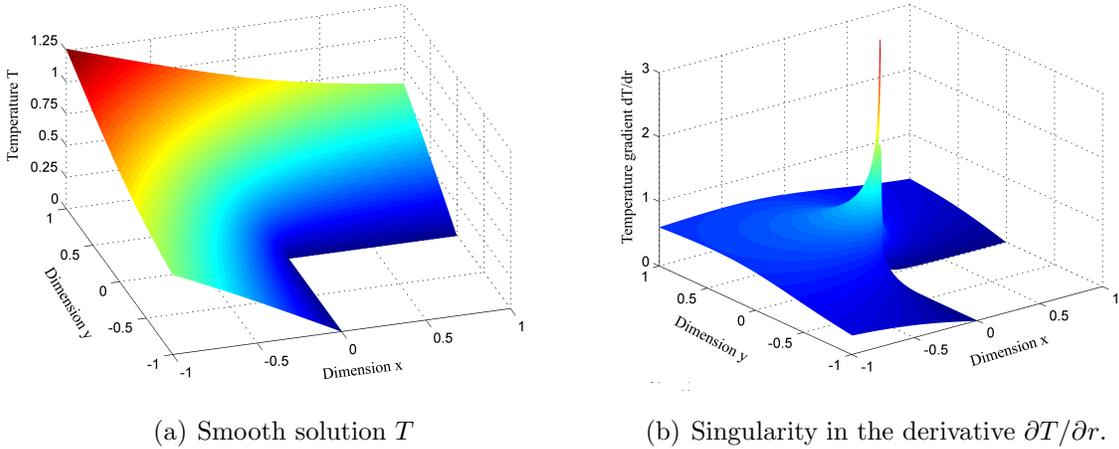


Figure 12: Heat conduction over the L-shaped domain with reentrant corner.

We discretize the L-shaped domain by two patches of  $10 \times 10$  NURBS elements. Subsequently, we add an increasing number of hierarchical levels around the reentrant corner by splitting half of the elements on the finest level in four. Figure 13a shows the adaptive NURBS mesh for the three-level case and Figure 13b depicts the corresponding set of quadrature points. As assumed in Section 3.4, the active functions on successive levels can overlap, however, functions on levels one and three, two and four, etc., do not.

In Figure 14 we observe that an increasing number of hierarchical levels improves the convergence rates by around one order of magnitude with respect to uniform refinement of the complete domain. We can clearly observe that the optimal full quadrature rules can successfully compete with full Gauss Legendre quadrature when it comes to accuracy. However, there is a significant reduction in quadrature points and assembly time.

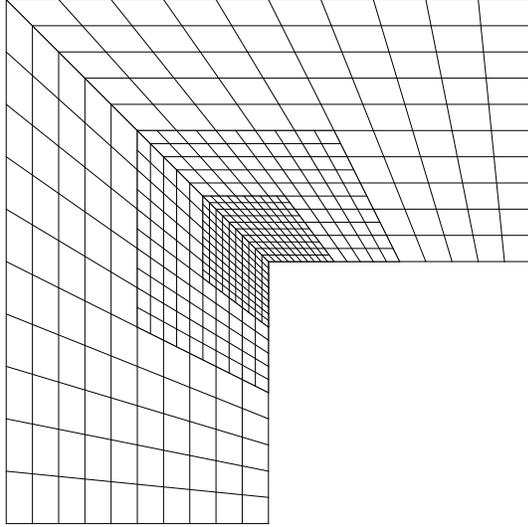
### 3.4.3 Fichera’s cube: 3d problem with a rough solution; multi-patch and adaptive local refinement

The second 3D test demonstrates the performance of the proposed quadrature rules for three-dimensional quadratic spline discretizations in a more complex setting, involving multiple patches, a non-smooth solution, and adaptive hierarchical spline refinement. It is defined by the model problem (16) over a cube with  $\alpha = 0$  and one excised octant (Fichera corner problem). The source term,

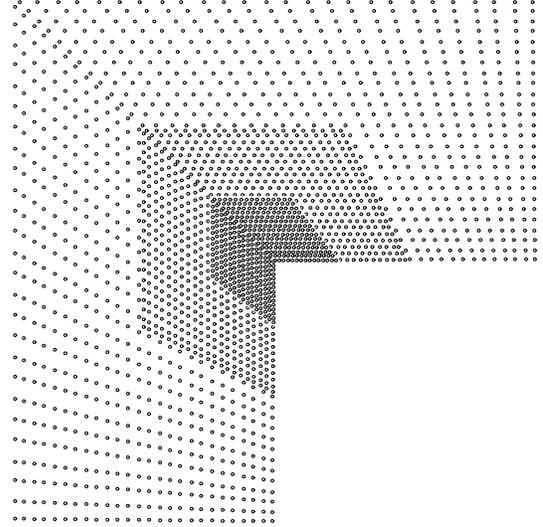
$$f = -\frac{3}{4} (x^2 + y^2 + z^2)^{-3/4}, \quad (29)$$

is manufactured in such a way that the exact solution to the model problem reads,

$$u = (x^2 + y^2 + z^2)^{1/4}, \quad (30)$$

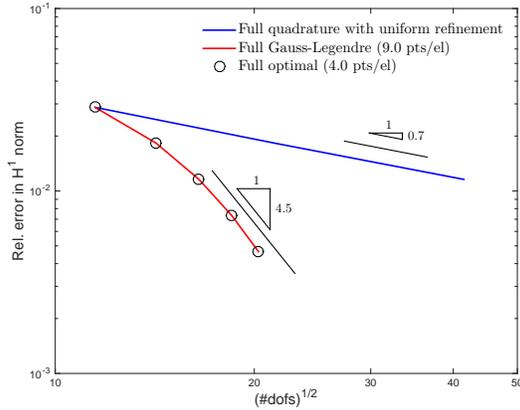


(a) Three-level hierarchical mesh refined towards the reentrant corner.

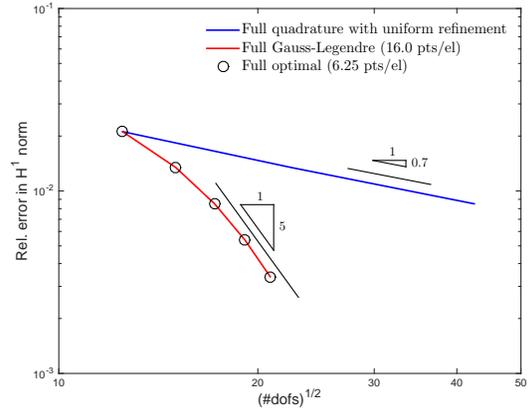


(b) Quadrature points corresponding to the three-level quadratic basis.

Figure 13: Adaptive isogeometric analysis: three-level hierarchical mesh and optimal quadrature point distribution for the L-shaped domain.



(a)  $p = 2$



(b)  $p = 3$

Figure 14: Convergence history under local hierarchical refinement versus the number of degrees of freedom for the L-shaped domain problem.

The origin of the coordinate system  $(x, y, z)$  is located at the reentrant corner. The exact solution (30) is not smooth, since it exhibits a singularity in the derivatives at the origin. Figure 15 illustrates the geometry and boundary conditions.

We discretize the domain by seven patches of quadratic B-splines, each of which covers one octant of the cube. The original mesh in each patch consists of  $5 \times 5 \times 5$  Bézier elements. We impose Dirichlet boundary conditions compatible with the exact solution (30) on all outer boundaries weakly by means of Nitsche's method. We examine the convergence behavior

using uniform refinement of all patches as well as adaptive refinement around the reentrant corner with hierarchical B-splines [12, 34, 42, 50]. The local refinement better accounts for the singularity in the gradient at the reentrant corner. Figure 16 shows the mesh after four hierarchical refinement steps. The corresponding solution field  $u$  and one of its derivatives,  $\partial u/\partial x$ , are plotted in Figures 17a and b, respectively.

Figure 18 plots the accuracy in terms of the relative error in the  $H^1$  semi-norm versus the number of degrees of freedom for uniform mesh refinement and local refinement through hierarchical B-splines. We compare results obtained using full quadrature of the optimal rule with full quadrature of the Gauss-Legendre rule. Results are not identical because the right-hand-side, which is non-polynomial, cannot be integrated exactly. The Gauss-Legendre rule is therefore slightly more accurate at the expense of a significant increase in quadrature points.

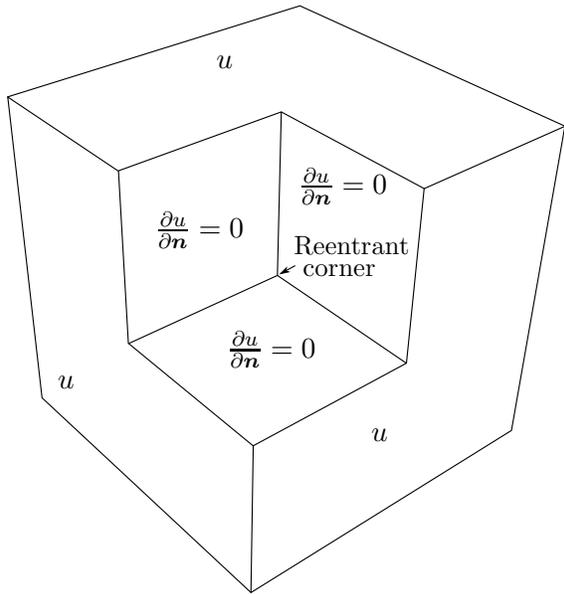


Figure 15: Fichera corner problem: Geometry and boundary conditions. The solution has a singularity in the gradient at the reentrant corner.

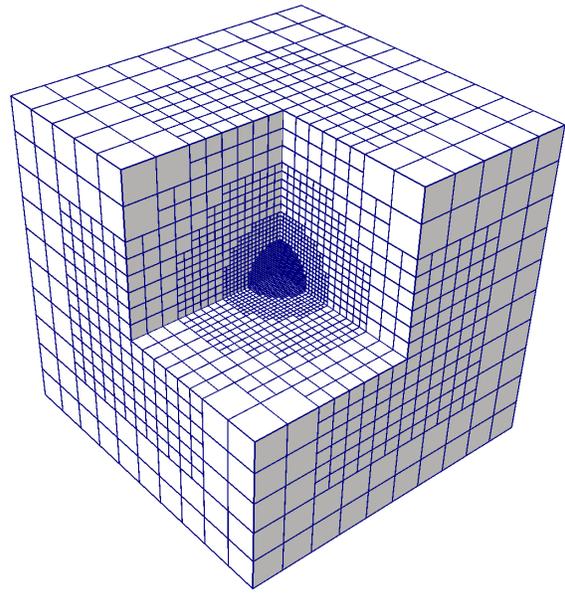
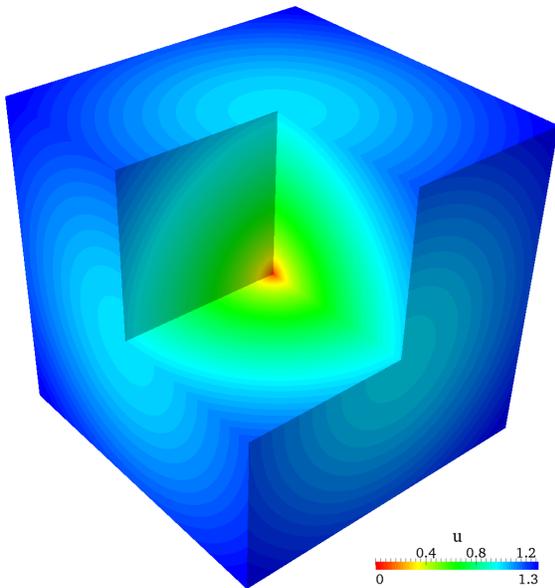
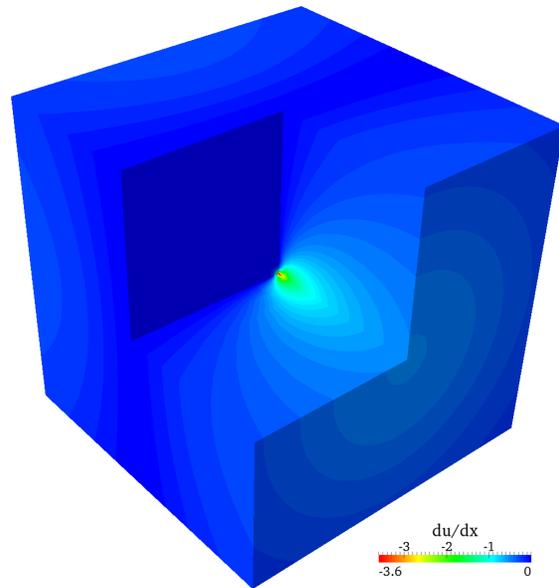


Figure 16: Hierarchically refined mesh of quadratic B-splines. The original mesh consists of seven patches, each of which covers one octant.



(a) Solution  $u$ .



(b) Solution gradient  $\partial u / \partial x$ .

Figure 17: Fichera corner problem: Solution of model problem and its gradient.

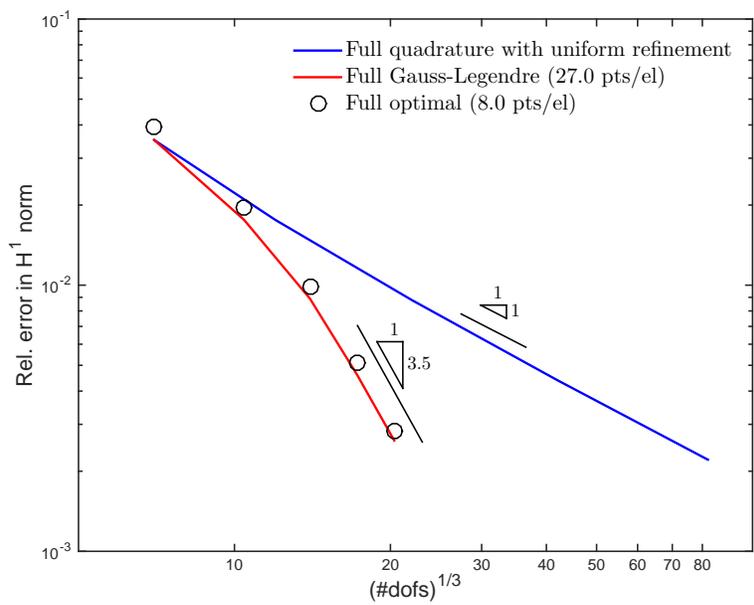


Figure 18:  $H^1$ -error under local hierarchical refinement versus the number of degrees of freedom for the Fichera's cube for a quadratic discretization.

## 4 Stability and accuracy of reduced quadrature

The reduced rules, introduced in the previous section, yielded results close to those obtained with full quadrature, while presenting another factor of two in computational savings. Special care is warranted, however, when applying these rules in practice. Reduced integration can, in general, lead to rank insufficient, or singular, stiffness matrices. In other words, the system can have eigenvalues, equal or close to zero, that are associated with spurious eigenmodes. In this section we show that the reduced rules proposed in this paper are stable and accurate. Application of these rules always yields a positive definite stiffness matrix, and hence a bounded, unique finite dimensional solution. By examining which polynomials add to the approximation power of the basis, and making sure their contribution to the bilinear form  $a(\cdot, \cdot)$  is exact, we can prove accuracy as well. More specifically, by “accuracy” we mean that the proposed reduced integration strategy results in an error,  $u^h - \tilde{u}^h$ , that is of the same order as the discretization error,  $u - u^h$  [10, 11]. To establish a final notion of stability of reduced integration, we carefully explore the numerical spectra of quadratic and cubic discretizations by way of eigenvalue analysis.

### 4.1 Conditions for stability and accuracy of reduced integration

The key question is: what degree of accuracy is required in the quadrature in order to maintain full rates of convergence? According to Strang & Fix [47]: “It is not required that every polynomial which appears be integrated exactly”. The reason for this is simply that not all polynomials in the trial space  $V^h$  contribute to the approximation power of the basis. This appears especially so in a tensor product setting. Figure 19 displays all monomials contained in a tensor product space of bi-degree 2 on a single two dimensional Bézier element. However, only the polynomials of total degree 2, that is,  $x^\alpha \cdot y^\beta$  for  $\alpha + \beta \leq 2$  featured in gray in Figure 19, contribute to the approximation power of the basis.

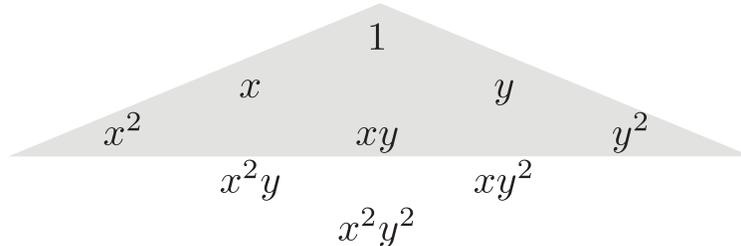


Figure 19: Only polynomials of total degree 2 (in gray) contribute to the approximation power of the basis. The additional monomials present in the tensor product space of bi-degree 2 in general do not.

Strang & Fix [47] present a simple theory that answers exactly the above stated question in the context of the classical finite element method. The basic idea, explained in the following example, is very similar to that of the patch test<sup>3</sup> [25, 47] and applies only to the leading terms in  $a^*(\cdot, \cdot)$ , involving first derivatives (case where  $\alpha = 0$ ).

---

<sup>3</sup>(31) is equivalent to satisfying the patch test in case  $P$  are linear polynomials.

**Example 4.1.** We consider a test and trial space  $V^h$  consisting of bi-quadratic splines defined on an affine domain. If  $a^*(\cdot, \cdot)$  is positive definite and

$$(a - a^*)(P, v^h) = 0 \quad \text{for all } P \in \mathcal{P}^2 \text{ and } v^h \in V^h. \quad (31)$$

on each Bézier element, then a positive constant  $C$  and a measure of the characteristic element size  $h$  exist such that the error due to quadrature is,

$$\|u^h - \tilde{u}^h\|_{H^1} \leq Ch^3. \quad (32)$$

At the element level the spaces  $V^h$  and  $\mathcal{P}^2$  consist of monomials displayed in Figures 20a and Figures 20b, respectively. The monomials in grey span the space of their derivatives and are the ones present in the bilinear form  $(a - a^*)(P^2, v^h)$ . Their product, which is displayed in dark grey in Figure 21, needs to be integrated exactly in order to maintain full approximation power. We urge the reader to compare this figure with Figure 5, which displays all polynomials contained in the corresponding tensor product reduced rule. The reduced rule clearly integrates exactly all polynomials required for optimal accuracy.

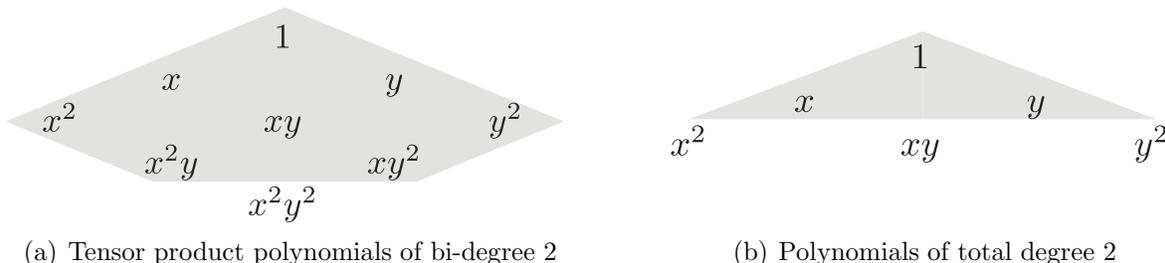


Figure 20: Monomials in the bi-quadratic space  $V^h$  and  $\mathcal{P}^2$  on a single Bézier element. The monomials in grey span the space of their derivatives and are the ones present in the bilinear form  $(a - a^*)(P^2, v^h)$  for  $\alpha = 0$ .

**Remark 4.1.** The error due to quadrature in (32) is bounded by the discretization error, which is order  $h^2$ . Hence, accuracy is not affected by reduced integration.

**Remark 4.2.** Positive definiteness of the bilinear form  $a^*(\cdot, \cdot)$  and the test in (31) are sufficient conditions in order to prove that application of a reduced quadrature rule is stable and accurate. However, numerical evidence [44] suggest that these conditions may not be necessary.

A proof of positive definiteness of the bilinear form  $a^*(\cdot, \cdot)$  is provided in the next section. A proof of accuracy, which requires some degree of regularity on material coefficients and the finite dimensional solution  $u^h$ , is discussed thereafter.

## 4.2 Stability

Stability of reduced integration demands that the bilinear form  $a^*(\cdot, \cdot)$  be positive definite, or equivalently that  $a^*(\cdot, \cdot)$  has positive eigenvalues. The basic test for definiteness is to determine whether or not there are non-zero test functions, which, under numerical quadrature, give up all their contribution to the bilinear form  $a^*(\cdot, \cdot)$ .

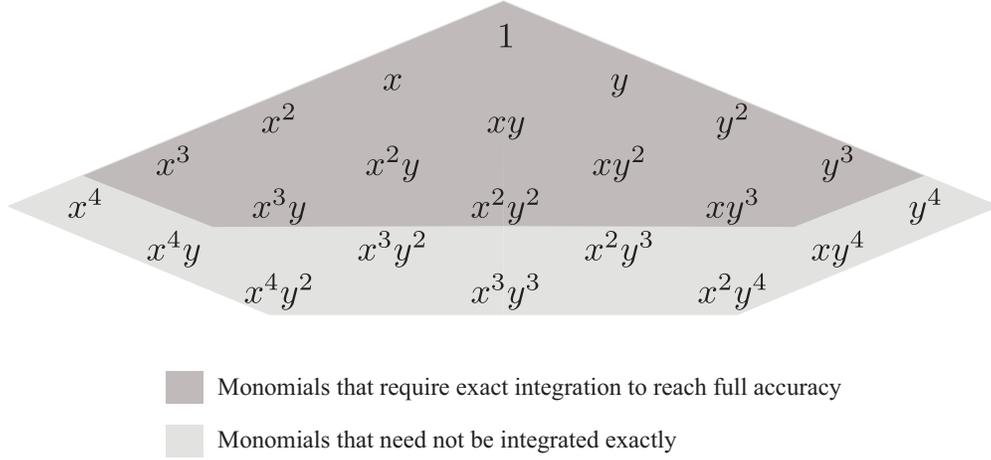


Figure 21: All monomials contained in  $a^*(P^2, v^h)$  for  $\alpha = 0$ . The monomials in dark grey need to be exactly integrated in order to maintain the same accuracy. The monomials in light grey need not be integrated exactly.

**Theorem 4.1.** *Let  $V^h := \mathcal{S}_r^p(\Omega)$  be a univariate spline space with zero boundary conditions and  $\{(\bar{\xi}_i, \bar{w}_i), i = 1, 2, \dots, m\}$  denote an  $m$ -point quadrature rule with positive weights associated with the target space  $\mathcal{S}_{r-1}^q(\Omega)$ , where  $q \geq p$ . Then,  $a^*(\cdot, \cdot)$ , resulting from quadrature of  $a(\cdot, \cdot)$  by the above rule, is positive definite, i.e. for  $v^h \in V^h$ ,*

$$a^*(v^h, v^h) \geq 0 \quad \text{and} \quad a^*(v^h, v^h) = 0 \implies v^h = 0. \quad (33)$$

*Proof.* Since  $\kappa, \alpha$  are positive we have that  $a^*(v^h, v^h) \geq 0$  for any quadrature rule with positive weights  $\bar{w}_i$ . To prove definiteness assume to the contrary that  $v^h \neq 0$  and that,

$$a^*(v^h, v^h) = \sum_{i=1}^m \bar{w}_i [\kappa(v_{,x}^h(\bar{\xi}_i))^2 + \alpha(v^h(\bar{\xi}_i))^2] = 0. \quad (34)$$

If  $q \geq p$ , then  $\mathcal{S}_r^p \subset \mathcal{S}_{r-1}^q$  and  $\mathcal{S}_{r-1}^{p-1} \subset \mathcal{S}_{r-1}^q$ . Consequently, the quadrature rule associated with target space  $\mathcal{S}_{r-1}^q(\Omega)$  exactly integrates  $v^h \in \mathcal{S}_r^p$  and its derivative  $v_{,x}^h \in \mathcal{S}_{r-1}^{p-1}$ . However, (34) implies that  $\{v^h(\bar{\xi}_i) = v_{,x}^h(\bar{\xi}_i) = 0\}_{i=1}^m$ , and hence, we must have that  $v^h = v_{,x}^h = 0$  everywhere. This contradiction completes the proof.  $\square$

**Remark 4.3.** *We note that, although positivity of the weights is not guaranteed by construction, we have never encountered optimal Gaussian quadrature rules resulting from our numerical algorithm that exhibit negative weights.*

**Remark 4.4.** *Extension to the multidimensional setting is straightforward and can be readily verified.*

### 4.3 Accuracy

In this section we analyze the accuracy of the optimal quadrature rule  $\{(\bar{\xi}_i, \bar{w}_i), i = 1, 2, \dots, m\}$  corresponding to the target space  $\mathcal{S}_r^q(\Omega)$ . In particular, we derive an expression for the residual due to approximate integration of a sufficiently smooth function  $g(\xi)$ . The estimates

obtained are subsequently used to determine the accuracy of reduced integration strategies applied to elliptic variational problems.

We define the following,

1.  $p \geq 1$  is a positive integer denoting the polynomial degree of the trial space.
2.  $\Omega = \{\xi_1, \xi_2, \dots, \xi_{n_{\text{el}}+1}\}$  denotes a partition of elements  $e_i = (\xi_i, \xi_{i+1})$ ,  $i = 1, 2, \dots, n_{\text{el}}$
3.  $\mathbf{r}$  denotes a vector prescribing the regularity,  $r_i \leq p - 1$ , at each knot  $\xi_i \in \Omega$ .
4.  $q \geq 2p - 1$  denotes the polynomial degree of the quadrature rule.
5.  $k \geq 0$  is a positive integer, and,

$$C_{\mathbf{r}}^k(\Omega) := \{g : g|_{e_i} \in C^k(\Omega) \text{ and } g^{(l)}(\xi_i^-) = g^{(l)}(\xi_i^+), l = 0, 1, \dots, r_i, \text{ for } i = 2, 3, \dots, n_{\text{el}}\}$$

**Remark 4.5.** *Because polynomials are  $C^\infty$  functions, note that  $\mathcal{S}_{\mathbf{r}}^q(\Omega) \subset C_{\mathbf{r}}^s(\Omega)$  for all  $s$ .*

**Remark 4.6.** *Also note that  $C_{\mathbf{r}}^{q+1}(\Omega) \subset C_{\mathbf{r}}^q(\Omega)$  and  $C_{\mathbf{r}}^q(\Omega) \subset C_{\mathbf{s}}^q(\Omega)$  if  $\mathbf{r} \geq \mathbf{s}$ , where the last inequality is intended component-wise.*

We will first show that with  $\mathcal{S}_{\mathbf{r}}^q(\Omega)$  we have good approximation of functions in  $C_{\mathbf{r}}^{q+1}(\Omega)$ . We will use the Newton representation of osculatory interpolating polynomials on  $q + 1$  knots,  $\{\tau_0 \leq \tau_1 \leq \dots \leq \tau_q\}$ , which uses the divided differences  $g[\tau_0, \dots, \tau_j]$  of a sufficiently smooth function  $g \in C^{q+1}([\tau_0, \tau_q])$ , see [20, Ch. I]. In particular, we recall that the degree  $q$  polynomial,

$$P_{\tau,g}(\xi) = \sum_{j=0}^q g[\tau_0, \dots, \tau_j] \prod_{k=0}^{j-1} (\xi - \tau_k),$$

interpolates function  $g$  at  $\{\tau_0 \leq \tau_1 \leq \dots \leq \tau_q\}$ , and, if knots are repeated, interpolates also the derivative, i.e.:

$$P_{\tau,g}^{(s)}(\tau_i) = g^{(s)}(\tau_i) \quad \text{for } s = 0, 1, \dots, r, \text{ if } \tau_i \text{ is repeated } r \text{ times.}$$

With this construction the following estimate holds [20, Ch. I, page 14]:

$$g(\xi) - P_{\tau,g}(\xi) = g[\tau_0, \dots, \tau_q, \xi] \prod_{k=0}^q (\xi - \tau_k) \quad \text{for all } \xi \in [\tau_0, \tau_q]. \quad (35)$$

We recall, see [20, Ch. IX], that the divided difference  $g[\tau_0, \dots, \tau_q, \xi]$  of function  $g \in C^{q+1}([\tau_0, \tau_q])$  admits the following representation,

$$g[\tau_0, \dots, \tau_q, \xi] = \frac{1}{(q+1)!} \int_{\tau_0}^{\tau_q} g^{(q+1)}(t) M(t, \xi) dt \quad (36)$$

where  $M(t, \xi)$  is the Peano kernel for the divided difference.

**Remark 4.7.**  $M(t, \xi)$  is actually a scaled B-spline function of degree  $q$  defined on the  $q + 2$  knots  $\{\tau_0, \dots, \tau_k, \xi, \tau_{k+1}, \dots, \tau_q\}$  with the property that  $\int_a^b M(t, \xi) dt = 1$ , for any  $\xi \in [\tau_0, \tau_q]$ , and  $0 \leq M(t, \xi) \leq \frac{q+1}{\tau_q - \tau_0}$ .

With these ingredients the following lemma provides an approximation  $g^h$  to  $g$  in a form that is useful for our future purpose.

**Lemma 4.1.** *Let  $g \in C_{\mathbf{r}}^{q+1}(\Omega)$  and  $h_i = \xi_{i+1} - \xi_i$ . There exists  $g^h \in \mathcal{S}_{\mathbf{r}}^q(\Omega)$  such that,*

$$|g(\xi) - g^h(\xi)| \leq \frac{h_i^q}{q!} \int_{\xi_i}^{\xi_{i+1}} |g^{(q+1)}(t)| dt, \quad \xi \in (\xi_i, \xi_{i+1}) \quad (37)$$

for all  $i = 1, 2, \dots, n_{\text{el}}$ .

*Proof.* For each element  $e_i = (\xi_i, \xi_{i+1})$ ,  $i = 1, 2, \dots, n_{\text{el}}$ , we consider the following construction. Consider a degree  $q$  osculatory polynomial  $P_{\tau^i, g}(\xi)$ , interpolating  $g \in C_{\mathbf{r}}^{q+1}(\Omega)$  at the following sequence of  $q + 1$  knots,

$$\left\{ \underbrace{\tau_0^i, \dots, \tau_{r_i}^i}_{r_i+1 \text{ times } \xi_i}, \dots, \tau_k^i, \tau_{k+1}^i, \dots, \underbrace{\tau_{q-r_{i+1}}^i, \dots, \tau_q^i}_{r_{i+1}+1 \text{ times } \xi_{i+1}} \right\}$$

We require  $r_i + r_{i+1} + 2 \leq q + 1$ , which is satisfied since  $r_i + r_{i+1} + 2 \leq 2p \leq q + 1$ .

By construction the successive polynomials  $P_{\tau^{i-1}, g}(\xi)$  and  $P_{\tau^i, g}(\xi)$  have  $r_i$  derivatives coinciding at  $\xi_i$ . Hence, there exists  $g^h \in \mathcal{S}_{\mathbf{r}}^q(\Omega)$  such that,

$$g^h|_{e_i} = P_{\tau^i, g}(\xi), \quad i = 1, 2, \dots, n_{\text{el}}$$

Because function  $g$  is sufficiently smooth we can use the estimate in (35) on each element, which combined with (36), the bound  $|M_i(t, \xi)| \leq \frac{q+1}{h_i}$ , and the inequality  $|\xi - \tau_k^i| \leq h_i$  for  $\xi \in (\xi_i, \xi_{i+1})$ , leads to the error estimate.  $\square$

**Remark 4.8.** *The construction of function  $g^h$  in the previous lemma is not unique, and depends on the remaining knots that were left unspecified.*

It is useful to write the result in Lemma 4.1 in a slightly different form. Consider the indicator function,

$$\chi_i(\xi) = \begin{cases} 1 & \text{if } \xi \in (\xi_i, \xi_{i+1}) \\ 0 & \text{otherwise} \end{cases}$$

Then, for any  $\xi \in [\xi_1, \xi_{n_{\text{el}}+1}]$ , (37) can be written as,

$$|g(\xi) - g^h(\xi)| \leq \sum_{i=1}^{n_{\text{el}}} \frac{h_i^q}{q!} \chi_i(\xi) \int_{\xi_i}^{\xi_{i+1}} |g^{(q+1)}(t)| dt \quad (38)$$

**Theorem 4.2.** Let  $g \in C_r^{q+1}(\Omega)$  and  $\{(\bar{\xi}_i, \bar{w}_i), i = 1, 2, \dots, m\}$  a quadrature rule that is exact for all functions in  $\mathcal{S}_r^q(\Omega)$ . Then there exists a positive constant  $C$  such that:

$$\left| \int_{\Omega} g(\xi) d\xi - \sum_{j=1}^m \bar{w}_j g(\bar{\xi}_j) \right| \leq Ch^{q+1} \sum_{i=1}^{n_{\text{el}}} \int_{e_i} |g^{(q+1)}(t)| dt,$$

where  $h = \max_i h_i$ .

*Proof.* Let  $g^h \in \mathcal{S}_r^q(\Omega)$  be constructed as in Lemma 4.1. Since  $g = (g - g^h) + g^h$  and using the fact that  $g^h$  is integrated exactly,

$$\left| \int_{\Omega} g(\xi) d\xi - \sum_{j=1}^m \bar{w}_j g(\bar{\xi}_j) \right| \leq \int_{\Omega} |g(\xi) - g^h(\xi)| d\xi + \sum_{j=1}^m |\bar{w}_j| |g(\bar{\xi}_j) - g^h(\bar{\xi}_j)|$$

Let  $J_i = \int_{\bar{\xi}_i}^{\xi_i+1} |g^{(q+1)}(t)| dt$ . Then we can bound each term as,

$$\begin{aligned} \int_{\Omega} |g(\xi) - g^h(\xi)| d\xi &\leq \int_{\Omega} \sum_{i=1}^{n_{\text{el}}} \frac{h_i^q}{q!} \chi_i(\xi) J_i d\xi = \sum_{i=1}^{n_{\text{el}}} \frac{h_i^q}{q!} J_i \int_{\Omega} \chi_i(\xi) d\xi \\ \sum_{j=1}^m |\bar{w}_j| \cdot |g(\bar{\xi}_j) - g^h(\bar{\xi}_j)| &\leq \sum_{j=1}^m |\bar{w}_j| \cdot \sum_{i=1}^{n_{\text{el}}} \frac{h_i^q}{q!} \chi_i(\bar{\xi}_j) J_i = \sum_{i=1}^{n_{\text{el}}} \frac{h_i^q}{q!} J_i \sum_{j=1}^m |\bar{w}_j| \chi_i(\bar{\xi}_j) \end{aligned}$$

We have that  $\int_{\Omega} \chi_i(\xi) d\xi = h_i$ , and similarly  $\sum_{j=1}^m |\bar{w}_j| \chi_i(\bar{\xi}_j) \propto h_i$ . Combining the bound on each term we obtain the result in the theorem.  $\square$

Theorem 4.2 provides a bound on the residual in quadrature of a function  $g \in C_r^{q+1}$ . In the following section we use this bound to determine the degree of accuracy,  $q$ , required to maintain optimal convergence rates in elliptic variational problems.

## 4.4 Accuracy of reduced quadrature applied to elliptic variational problems

We recall that  $a(\cdot, \cdot)$  and  $l(\cdot)$  arose from exact integration of the Galerkin equations modeling reaction-diffusion, and  $a^*(\cdot, \cdot)$  and  $l^*(\cdot)$  denoted their numerical approximation by quadrature. Furthermore,  $u^h$  and  $\tilde{u}^h$  denoted the numerical solution to  $a(u^h, v^h) = l(v^h)$  and  $a^*(\tilde{u}^h, v^h) = l^*(v^h)$ , respectively. The key to successful reduced integration is to bound the error due to quadrature,  $u^h - \tilde{u}^h$ , by that of the discretization error,  $u - u^h$ . We will need the following lemma.

**Lemma 4.2.** Let  $u^h, v^h \in \mathcal{S}_r^p(\Omega)$  and  $a^*(\cdot, \cdot)$  and  $l^*(\cdot)$  denote the approximation of  $a(\cdot, \cdot)$  and  $l(\cdot)$  by means of the quadrature rule  $\{(\bar{\xi}_i, \bar{w}_i), i = 1, 2, \dots, m\}$ , which is exact for all functions in  $\mathcal{S}_{r-1}^q(\Omega)$ . Then there exists a constant  $C$  and a measure of the maximum element size  $h$  such that,

$$|(a^* - a)(u^h, v^h)| + |(l^* - l)(v^h)| \leq Ch^{q-p+2} \|v^h\|_{H^1}$$

*Proof.* Let  $D^{(k)} := \frac{d^k}{d\xi^k}$  be shorthand for the  $k$ 'th derivative and consider functions  $g_1$  and  $g_2$  in  $C_{\mathbf{r}-1}^{q+1}$ . Their product is also in  $C_{\mathbf{r}-1}^{q+1}$ . Therefore, according to Theorem 4.2,

$$\left| \int_{\Omega} g_1(\xi)g_2(\xi)d\xi - \sum_{j=1}^m \bar{w}_j g_1(\bar{\xi}_j)g_2(\bar{\xi}_j) \right| \leq Ch^{q+1} \sum_{i=1}^{n_{\text{el}}} \int_{e_i} |D^{(q+1)}(g_1g_2)| d\xi \quad (39)$$

Each integral on the right-hand-side corresponding to element  $e_i$  can be written as,

$$\begin{aligned} \|D^{(q+1)}(g_1g_2)\|_{L^1(e_i)} &= \left\| \sum_{k=0}^{q+1} c_k D^{(q+1-k)}g_1 \cdot D^{(k)}g_2 \right\|_{L^1(e_i)} && \text{(product rule of differentiation)} \\ &\leq \sum_{k=0}^{q+1} c_k \|D^{(q+1-k)}g_1 \cdot D^{(k)}g_2\|_{L^1(e_i)} && \text{(Triangle inequality)} \\ &\leq \sum_{k=0}^{q+1} c_k \|D^{(q+1-k)}g_1\|_{L^2(e_i)} \cdot \|D^{(k)}g_2\|_{L^2(e_i)} && \text{(Cauchy Schwartz)} \end{aligned}$$

where  $c_k = \binom{q+1}{k}$ .

Since  $g_1 \in C_{\mathbf{r}-1}^{q+1}$ , all its  $q+1$  derivatives exist in element  $e_i$ . Therefore,  $\|D^{(q+1-k)}g_1\|_{L^2(e_i)}$  is bounded for  $k = 0, 1, \dots, q+1$ . In the following  $g_2$  is a piecewise polynomial of degree  $t$ . Differentiation beyond  $k = t$  will return zero, therefore,

$$\|D^{(q+1)}(g_1g_2)\|_{L^1(e_i)} \leq \sum_{k=0}^t C_k \|D^{(k)}g_2\|_{L^2(e_i)}$$

where  $C_k$ ,  $k = 0, 1, \dots, t$ , are positive and bounded constants. By means of the inverse estimate  $\|D^{(k+1)}g_2\| \leq \bar{C}h^{-1}\|D^{(k)}g_2\|$ , where  $\bar{C}$  is a positive constant, we are able to bound derivatives at the expense of factors of  $h^{-1}$ .

Then we consider the following cases,

1. Estimate  $|(a^* - a)(u^h, v^h)|$  with  $\kappa \neq 0$  and  $\alpha = 0$ : set  $g_1 = u_{,x}^h \in \mathcal{S}_{\mathbf{r}-1}^{p-1} \subset C_{\mathbf{r}-1}^{q+1}$  and  $g_2 = v_{,x}^h \in \mathcal{S}_{\mathbf{r}-1}^{p-1} \subset C_{\mathbf{r}-1}^{q+1}$ . Both are of polynomial degree  $t = p-1$ , so we obtain the estimate,

$$\begin{aligned} |(a^* - a)(u^h, v^h)| &\leq Ch^{q+1} \sum_{i=1}^{n_{\text{el}}} (C_0 \|v_{,x}^h\|_{L^2(e_i)} + C_1 \|Dv_{,x}^h\|_{L^2(e_i)} + \dots + C_{p-1} \|D^{p-1}v_{,x}^h\|_{L^2(e_i)}) \\ &\leq Ch^{q+1} \sum_{i=1}^{n_{\text{el}}} \bar{C}_0 h^{-(p-1)} \|v_{,x}^h\|_{L^2(e_i)} \\ &\leq Ch^{q-p+2} \|v^h\|_{H^1} \end{aligned}$$

2. Estimate  $|(a^* - a)(u^h, v^h)|$  with  $\kappa = 0$  and  $\alpha \neq 0$ : set  $g_1 = u^h \in \mathcal{S}_{\mathbf{r}}^p \subset C_{\mathbf{r}-1}^{q+1}$  and

$g_2 = v^h \in \mathcal{S}_r^p \subset C_{r-1}^{q+1}$ . Both are of polynomial degree  $t = p$ , so we obtain the estimate,

$$\begin{aligned} |(a^* - a)(u^h, v^h)| &\leq Ch^{q+1} \sum_{i=1}^{n_{el}} (C_0 \|v^h\|_{L^2(e_i)} + C_1 \|Dv^h\|_{L^2(e_i)} + \dots + C_p \|D^p v^h\|_{L^2(e_i)}) \\ &\leq Ch^{q+1} \sum_{i=1}^{n_{el}} (C_0 \|v^h\|_{L^2(e_i)} + \bar{C}_1 h^{-(p-1)} \|v_{,x}^h\|_{L^2(e_i)}) \\ &\leq Ch^{q-p+2} \|v^h\|_{H^1} \end{aligned}$$

3. Estimate  $|(l^* - l)(v^h)|$ : set  $g_1 = f \in C_{r-1}^{q+1}$  and  $g_2 = v^h \in \mathcal{S}_r^p \subset C_{r-1}^{q+1}$ , which is of polynomial of degree  $t = p$ . Then we obtain the estimate,

$$\begin{aligned} |(l^* - l)(v^h)| &\leq Ch^{q+1} \sum_{i=1}^{n_{el}} (C_0 \|v^h\|_{L^2(e_i)} + C_1 \|Dv^h\|_{L^2(e_i)} + \dots + C_p \|D^p v^h\|_{L^2(e_i)}) \\ &\leq Ch^{q+1} \sum_{i=1}^{n_{el}} (C_0 \|v^h\|_{L^2(e_i)} + \bar{C}_1 h^{-(p-1)} \|v_{,x}^h\|_{L^2(e_i)}) \\ &\leq Ch^{q-p+2} \|v^h\|_{H^1} \end{aligned}$$

Combining the three estimates proves the lemma.  $\square$

The following theorem from Strang & Fix [47] provides sufficient conditions for accurate reduced integration.

**Theorem 4.3.** *Suppose that  $a^*(\cdot, \cdot)$  is positive definite, i.e.  $a^*(v^h, v^h) \geq \theta \|v^h\|_{H^1}^2$  and that,*

$$|(a^* - a)(u^h, v^h)| + |(l^* - l)(v^h)| \leq Ch^{q-p+2} \|v^h\|_{H^1} \quad (40)$$

*Then a positive constant  $C$  and a measure of the characteristic element size  $h$  exist such that the error between  $\tilde{u}^h$  and  $u^h$  due to approximate integration is of order,*

$$\|u^h - \tilde{u}^h\|_{H^1} \leq Ch^{q-p+2}. \quad (41)$$

*Proof.* Using  $a(u^h, v^h) = l(v^h)$  and  $a^*(\tilde{u}^h, v^h) = l^*(v^h)$ , positive definiteness of  $a^*(\cdot, \cdot)$  and the bound obtained above we obtain,

$$\begin{aligned} \theta \|u_h - \tilde{u}^h\|_{H^1}^2 &\leq a^*(u^h - \tilde{u}^h, u^h - \tilde{u}^h) \\ &= a^*(u^h, u^h - \tilde{u}^h) - a^*(\tilde{u}^h, u^h - \tilde{u}^h) \\ &= (a^* - a)(u^h, u^h - \tilde{u}^h) - (l^* - l)(u^h - \tilde{u}^h) \\ &\leq |(a^* - a)(u^h, u^h - \tilde{u}^h)| + |(l^* - l)(u^h - \tilde{u}^h)| \leq Ch^{q-p+2} \|u_h - \tilde{u}^h\|_{H^1} \end{aligned}$$

Deviding by the common factor on both sides and we obtain (41).  $\square$

The reduced integration rules we have proposed in this paper are of degree  $q = 2p - 1$ . Consequently, the error due to approximate integration,  $\|u^h - \tilde{u}^h\|_{H^1} \sim \mathcal{O}(h^{p+1})$ , is of higher order than the discretization error,  $\|u - u^h\|_{H^1} \sim \mathcal{O}(h^p)$ .

**Remark 4.9.** *The results obtained in the univariate setting in Sections 4.2 - 4.4, can easily be extended to the multidimensional tensor product setting, see Stroud [48, Ch. 4, §4.5]. For extension to mapped geometries within the isoparametric paradigm we refer to Ciarlet [16, Ch. 4, §4.4].*

## 4.5 Analysis of discrete spectra under reduced integration

In the previous analysis we have shown that the discrete set of equations resulting from our reduced integration formulae are rank sufficient, thereby excluding spurious zero-energy modes. To establish a final notion of stability, we need to show in addition to rank sufficiency that the reduced quadrature schemes do not lead to spurious modes with finite energy, that is, there are no non-zero spurious eigenvalues within the discrete spectrum. We therefore analyze the accuracy of the discrete spectrum, that is, the set of eigenvalues resulting from the numerical solution of a generalized eigenvalue problem. In contrast to standard error estimation, discrete spectrum analysis enables us to analyze the approximation properties of all the scales present in a discretization scheme [21, 25, 28]. In the context of this work, we are particularly interested in the effect of reduced quadrature on the accuracy of the discrete spectrum. While not a proof, such a study does give us insight into the numerical approximation made over the whole range of modes and hence can give us the confidence we need to apply these rules to similar problems, including ones involving more complicated geometry.

### 4.5.1 A generalized eigenvalue problem

Let  $\Omega := (0, 1)^2 \subset \mathbb{R}^2$  and  $\Gamma$  its boundary, and consider the eigenvalue problem:

$$-\Delta\phi = \lambda\phi \quad \text{in } \Omega \tag{42a}$$

$$\phi = 0 \quad \text{on } \Gamma \tag{42b}$$

where a solution  $(\lambda, u)$  is an eigenvalue, eigenvector pair.

Transforming (42) into the equivalent weak formulation and discretizing using the Galerkin method we obtain the discrete problem: find  $(\lambda^h, \phi^h) \in \mathbb{R}^+ \times V^h$  such that for all  $v^h \in V^h$ ,

$$\int_{\Omega} \nabla v^h \cdot \nabla \phi^h \, d\Omega = \lambda^h \int_{\Omega} v^h \phi^h \, d\Omega \tag{43}$$

This is a generalized eigenvalue problem, which can be represented in matrix notation as: find  $\{(\lambda_k^h, \phi_k) \in \mathbb{R} \times \mathbb{R}^n, k = 1, \dots, n\}$  such that,

$$K\phi_k = \lambda_k^h M\phi_k. \tag{44}$$

Here  $M$  and  $K$  denote the Galerkin mass and stiffness matrix, respectively, which are both symmetric and positive definite.

For simplicity, we assume that the parametric space  $\Omega'$  and physical space  $\Omega$  coincide. In this case the nonzero entries of the mass and stiffness matrices can be evaluated by numerical quadrature as,

$$M_{ij} = \int_{\Omega} N_i N_j d\Omega \approx \sum_{k=1}^m N_i(\bar{\boldsymbol{\xi}}_k) N_j(\bar{\boldsymbol{\xi}}_k) \bar{w}_k \quad (45)$$

$$K_{ij} = \int_{\Omega} \nabla N_i \cdot \nabla N_j d\Omega \approx \sum_{k=1}^m \nabla N_i(\bar{\boldsymbol{\xi}}_k) \cdot \nabla N_j(\bar{\boldsymbol{\xi}}_k) \bar{w}_k \quad (46)$$

where  $\{(\bar{\boldsymbol{\xi}}_k, \bar{w}_k), k = 1, 2, \dots, m\}$  denote the quadrature points and weights of the multi-dimensional tensor product quadrature rule and  $\{N_j, j = 0, 1, \dots, n\}$  the multidimensional B-spline basis functions.

#### 4.5.2 Spectral decomposition of the stiffness matrix

For the approximation power of any discretization scheme, the accuracy of the lower part of the discrete spectrum is crucial. To illustrate this statement for an elliptic boundary-value problem, we briefly review a few identities on the spectral decomposition of the stiffness matrix. Let us consider the discrete Laplace problem  $K\mathbf{u} = \mathbf{f}$ , where  $K$  and  $\mathbf{f}$  denote the stiffness matrix and the force vector, and  $\mathbf{u}$  is the vector of unknowns. We can expand  $\mathbf{u}$  in terms of the eigenmodes as,

$$\mathbf{u} = \sum_{k=1}^n \boldsymbol{\phi}_k c_k. \quad (47)$$

Using orthogonality of the modes  $\boldsymbol{\phi}_k$ , it can be shown that the coefficients  $c_k, k = 1, \dots, n$  are determined by the relation  $\lambda_k^h \boldsymbol{\phi}_k^T M \boldsymbol{\phi}_k c_k = \boldsymbol{\phi}_k^T \mathbf{f}$ , and therefore,

$$\mathbf{u} = \left( \sum_{k=1}^n \frac{1}{\lambda_k^h} \frac{\boldsymbol{\phi}_k \boldsymbol{\phi}_k^T}{\boldsymbol{\phi}_k^T M \boldsymbol{\phi}_k} \right) \mathbf{f} \quad (48)$$

The term in brackets can be identified as the inverse of the stiffness matrix,  $K^{-1}$ , in spectrally decomposed form. Each component  $k$  of the sum represents the contribution of the corresponding eigenmode to the inverse and  $1/\lambda_k^h$  is its participation factor. Since  $\lambda_k^h, k = 1, 2, \dots$  is a monotonically increasing sequence, the contributions of the lowest eigenmodes,  $k = 1, 2, 3, 4, \dots$  will typically dominate the solution  $\mathbf{u}$ . We remark that such an analysis holds for the case of elliptic boundary-value problems, but not for hyperbolic initial boundary-value problems, such as ones arising in elastodynamics and structural dynamics [27].

#### 4.5.3 Discrete spectra

In the following, we consider the generalized eigenvalue problem for the 2D Laplace operator. Formulae for the exact eigenvalues and eigenmodes of the continuous eigenvalue problem are

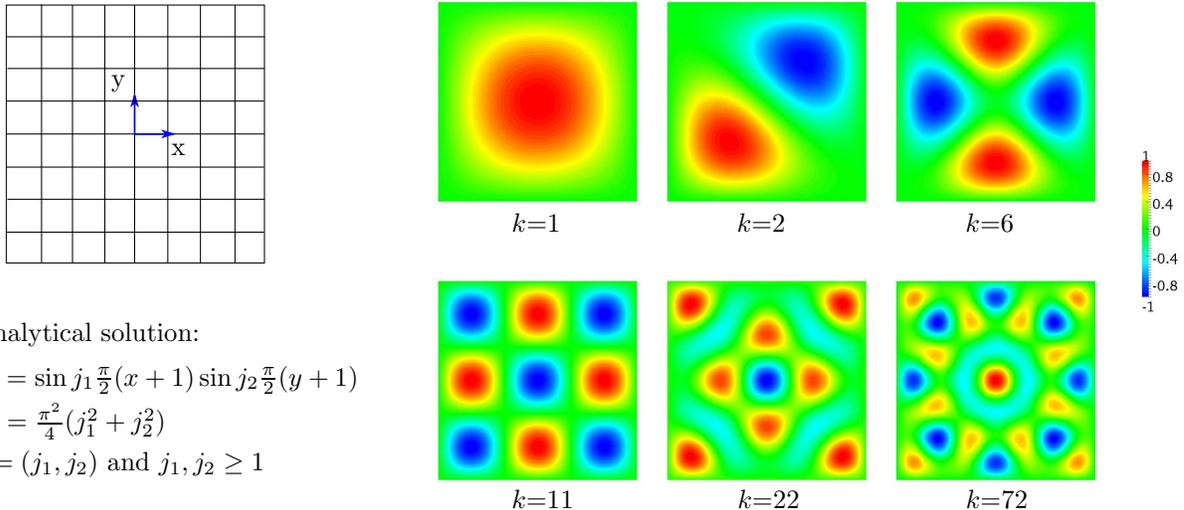


Figure 22: The generalized eigenvalue problem for the Laplace operator on the square domain  $\Omega = (-1, 1)^2$  with homogeneous Dirichlet boundary conditions: Structured B-spline discretization, exact eigenvalues and eigenmodes, and some numerical mode shapes of mode number  $k$ .

given in Figure 22. We have computed the discrete spectrum of eigenvalues and corresponding mode shapes using one-patch, maximally smooth quadratic and cubic uniform spline discretizations of  $50 \times 50$  Bézier elements.

We compare the numerically computed eigenvalues,  $\lambda_k^h$ ,  $k = 1, 2, \dots$ , of the discrete eigenvalue problem with the analytically computed eigenvalues,  $\lambda_k$ ,  $k = 1, 2, \dots$ , of the continuous eigenvalue problem. To this end, we plot the ratio of eigenvalues,  $\lambda_k^h/\lambda_k$ , versus the mode number  $k$ , normalized by the total number of degrees of freedom  $n_{eq}$ . The closer the ratio of eigenvalues to one, the better is the accuracy of the discrete spectrum.

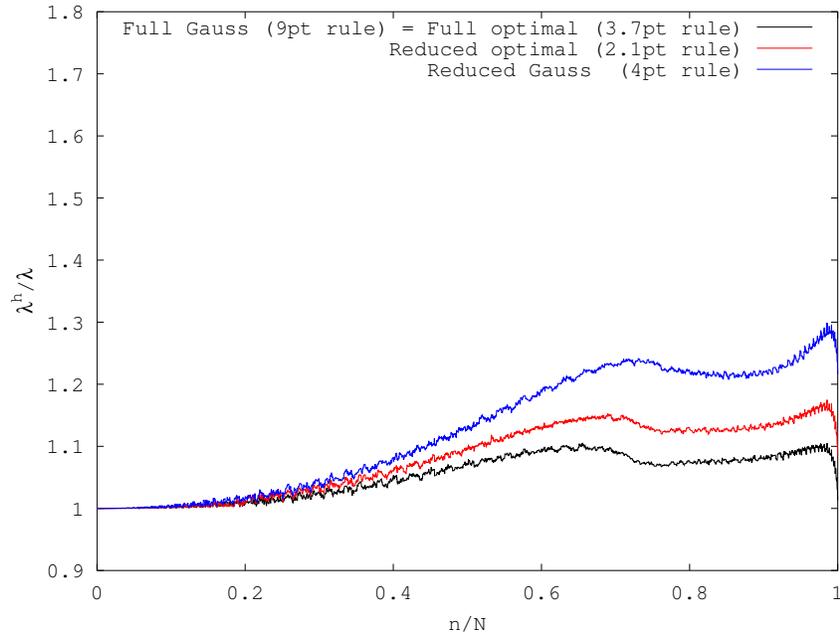
Figures 23a and 23b depict the resulting normalized discrete spectra for the quadratic and cubic spline discretizations, respectively, where both stiffness and mass matrices are integrated by the same quadrature rule. The full quadrature of the optimal and Gauss-Legendre rule lead to identical results and are depicted in black. The results obtained with the reduced rules are shown in red and may be compared with reduced Gauss-rules (blue). The lowest modes of the discrete spectrum determine the asymptotic accuracy of the approximation. Those computed with the reduced rules exhibit similar accuracy to the low modes obtained with full Gauss-Legendre quadrature.

We have observed that the optimal reduced rule for both quadratic and cubic discretizations lead to well-behaved eigenvalues throughout the complete discrete spectrum in the sense that there are no spurious finite-energy modes with non-zero eigenvalues in the discrete spectrum. Tables 7 and 8 depict the first eighty normalized eigenvalues obtained with reduced integration corresponding to the quadratic and cubic discretizations, respectively. Such an analysis has been performed for the entire spectrum and no spurious modes have been observed. In conjunction with rank sufficiency, which has been shown previously in this section, this observation empirically supports a notion of stability for quadratic and cubic spline discretizations with quadrature based on the proposed reduced rules. We conclude

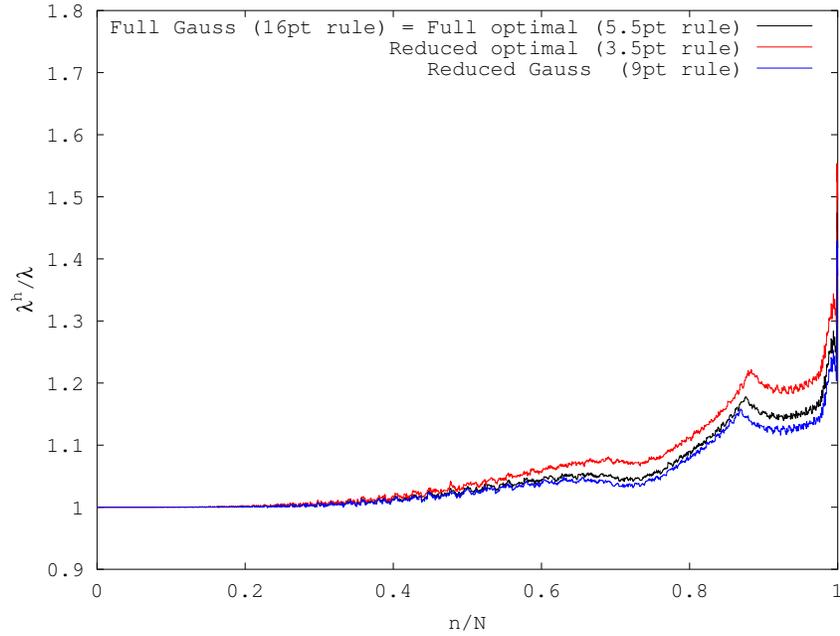
2D eigenvalue problem: numerical eigenvalues $\lambda_k^h$ with the reduced rule corresponding to $\mathcal{S}_0^3 \otimes \mathcal{S}_0^3$ versus analytical eigenvalues $\lambda_k$ . No spurious modes are reported.								
$k$	$\lambda_k^h$	$\lambda_k$	$\lambda_k^h/\lambda_k$		$k$	$\lambda_k^h$	$\lambda_k$	$\lambda_k^h/\lambda_k$
1	4.9348	4.9348	1.00000		41	150.5115	150.5110	0.999997
2	12.3370	12.3370	1.00000		42	160.3811	160.3810	1.000000
3	12.3370	12.3370	1.00000		43	160.3811	160.3810	1.000000
4	19.7392	19.7392	1.00000		44	160.3811	160.3810	1.000000
5	24.6740	24.6740	1.00000		45	160.3811	160.3810	1.000000
6	24.6740	24.6740	1.00000		46	167.7833	167.7830	0.999998
7	32.0762	32.0762	1.00000		47	167.7833	167.7830	0.999998
8	32.0762	32.0762	1.00000		48	177.6529	177.6530	1.000001
9	41.9458	41.9458	1.00000		49	180.1203	180.1200	0.999998
10	41.9458	41.9458	1.00000		50	180.1203	180.1200	0.999998
11	44.4132	44.4132	1.00000		51	182.5877	182.5880	1.000002
12	49.3480	49.3480	1.00000		52	182.5877	182.5880	1.000002
13	49.3480	49.3480	1.00000		53	197.3921	197.3920	1.000000
14	61.6850	61.6850	1.00000		54	197.3921	197.3920	1.000000
15	61.6850	61.6850	1.00000		55	202.3269	202.3270	1.000001
16	64.1524	64.1524	1.00000		56	202.3269	202.3270	1.000001
17	64.1524	64.1524	1.00000		57	209.7291	209.7290	1.000000
18	71.5546	71.5546	1.00000		58	209.7291	209.7290	1.000000
19	71.5546	71.5546	1.00000		59	209.7291	209.7290	1.000000
20	78.9568	78.9568	1.00000		60	209.7291	209.7290	1.000000
21	83.8916	83.8916	1.00000		61	219.5987	219.5990	1.000001
22	83.8916	83.8916	1.00000		62	219.5987	219.5990	1.000001
23	91.2938	91.2938	1.00000		63	222.0661	222.0660	1.000000
24	91.2938	91.2938	1.00000		64	222.0661	222.0660	1.000000
25	98.6960	98.6960	1.00000		65	239.3379	239.3380	1.000000
26	98.6960	98.6960	1.00000		66	239.3379	239.3380	1.000000
27	101.1634	101.1630	1.00000		67	241.8053	241.8050	0.999999
28	101.1634	101.1630	1.00000		68	246.7401	246.7400	1.000000
29	111.0330	111.0330	1.00000		69	246.7401	246.7400	1.000000
30	111.0330	111.0330	1.00000		70	249.2075	249.2080	1.000002
31	123.3701	123.3700	1.00000		71	249.2075	249.2080	1.000002
32	123.3701	123.3700	1.00000		72	256.6097	256.6100	1.000001
33	123.3701	123.3700	1.00000		73	256.6097	256.6100	1.000001
34	128.3049	128.3050	1.00000		74	261.5445	261.5450	1.000002
35	128.3049	128.3050	1.00000		75	261.5445	261.5450	1.000002
36	130.7723	130.7720	1.00000		76	268.9467	268.9470	1.000001
37	130.7723	130.7720	1.00000		77	268.9467	268.9470	1.000001
38	143.1093	143.1090	1.00000		78	278.8163	278.8160	0.999999
39	143.1093	143.1090	1.00000		79	278.8163	278.8160	0.999999
40	150.5115	150.5110	1.00000		80	286.2185	286.2190	1.000002

Table 7: List of lowest eigenvalues for the 2D Laplace eigenvalue problem with homogeneous Dirichlet boundary conditions. The  $\lambda_k^h$ , computed from  $50 \times 50$  quadratic Bézier elements with the reduced rule corresponding to  $\mathcal{S}_0^3 \otimes \mathcal{S}_0^3$ , are compared to the exact eigenvalues  $\lambda_k$  of the continuous problem.

that these quadrature rules need no further stabilizing mechanisms, although we do not have a mathematical proof of this.



(a) Quadratic spline discretization



(b) Cubic spline discretization

Figure 23: Comparison of the normalized discrete spectrum of the Laplace operator with homogeneous Dirichlet boundary conditions, computed on a  $50 \times 50$  mesh with different quadrature rules.

2D case: Numerical eigenvalues $\lambda_k^h$ with the reduced rule corresponding to $\mathcal{S}_1^5 \otimes \mathcal{S}_1^5$ versus analytical eigenvalues $\lambda_k$ . No spurious modes are reported.								
$k$	$\lambda_k^h$	$\lambda_k$	$\lambda_k^h/\lambda_k$		$k$	$\lambda_k^h$	$\lambda_k$	$\lambda_k^h/\lambda_k$
1	4.9348	4.9348	1.00000		41	150.5115	150.5110	0.999997
2	12.3370	12.3370	1.00000		42	160.3811	160.3810	1.000000
3	12.3370	12.3370	1.00000		43	160.3811	160.3810	1.000000
4	19.7392	19.7392	1.00000		44	160.3811	160.3810	1.000000
5	24.6740	24.6740	1.00000		45	160.3811	160.3810	1.000000
6	24.6740	24.6740	1.00000		46	167.7833	167.7830	0.999998
7	32.0762	32.0762	1.00000		47	167.7833	167.7830	0.999998
8	32.0762	32.0762	1.00000		48	177.6529	177.6530	1.000001
9	41.9458	41.9458	1.00000		49	180.1203	180.1200	0.999998
10	41.9458	41.9458	1.00000		50	180.1203	180.1200	0.999998
11	44.4132	44.4132	1.00000		51	182.5877	182.5880	1.000002
12	49.3480	49.3480	1.00000		52	182.5877	182.5880	1.000002
13	49.3480	49.3480	1.00000		53	197.3921	197.3920	1.000000
14	61.6850	61.6850	1.00000		54	197.3921	197.3920	1.000000
15	61.6850	61.6850	1.00000		55	202.3269	202.3270	1.000001
16	64.1524	64.1524	1.00000		56	202.3269	202.3270	1.000001
17	64.1524	64.1524	1.00000		57	209.7291	209.7290	1.000000
18	71.5546	71.5546	1.00000		58	209.7291	209.7290	1.000000
19	71.5546	71.5546	1.00000		59	209.7291	209.7300	1.000004
20	78.9568	78.9568	1.00000		60	209.7291	209.7300	1.000004
21	83.8916	83.8916	1.00000		61	219.5987	219.5990	1.000001
22	83.8916	83.8916	1.00000		62	219.5987	219.5990	1.000001
23	91.2938	91.2939	1.00000		63	222.0661	222.0670	1.000004
24	91.2938	91.2939	1.00000		64	222.0661	222.0670	1.000004
25	98.6960	98.6961	1.00000		65	239.3379	239.3380	1.000000
26	98.6960	98.6961	1.00000		66	239.3379	239.3380	1.000000
27	101.1634	101.1630	1.00000		67	241.8053	241.8050	0.999999
28	101.1634	101.1630	1.00000		68	246.7401	246.7400	1.000000
29	111.0330	111.0330	1.00000		69	246.7401	246.7400	1.000000
30	111.0330	111.0330	1.00000		70	249.2075	249.2090	1.000006
31	123.3701	123.3700	1.00000		71	249.2075	249.2090	1.000006
32	123.3701	123.3700	1.00000		72	256.6097	256.6110	1.000005
33	123.3701	123.3700	1.00000		73	256.6097	256.6110	1.000005
34	128.3049	128.3050	1.00000		74	261.5445	261.5450	1.000002
35	128.3049	128.3050	1.00000		75	261.5445	261.5450	1.000002
36	130.7723	130.7720	1.00000		76	268.9467	268.9480	1.000005
37	130.7723	130.7720	1.00000		77	268.9467	268.9480	1.000005
38	143.1093	143.1090	1.00000		78	278.8163	278.8170	1.000002
39	143.1093	143.1090	1.00000		79	278.8163	278.8170	1.000002
40	150.5115	150.5110	1.00000		80	286.2185	286.2200	1.000005

Table 8: List of lowest eigenvalues for the 2D Laplace eigenvalue problem with homogeneous Dirichlet boundary conditions. The  $\lambda_k^h$ , computed from  $50 \times 50$  cubic Bézier elements with the reduced rule corresponding to  $\mathcal{S}_1^5 \otimes \mathcal{S}_1^5$ , are compared to the exact eigenvalues  $\lambda_k$  of the continuous problem.

## 5 Conclusions

We continued the study initiated in [29] in search of optimal quadrature rules for tensor product isogeometric analysis. Therein a Newton method was presented in order to solve the governing system of non-linear equations, and several rules of practical interest were obtained. However, these were limited to spline spaces of modest degree, continuity, and dimension due to the highly non-linear behavior of the equations. Furthermore, ill-conditioning limited the accuracy of the rules below the preferred 16 digits of accuracy.

We have presented a variety of techniques to mitigate these difficulties. The first were an improved starting guess and a continuation algorithm in order to consistently produce accurate results. The proposed starting guess, obtained by solving a series of localized non-linear problems, is highly accurate and in certain cases exact. Our approach to continuation is very simple and is based on the residual of the the non-linear equations. We employed arbitrary precision arithmetic in order to compute quadrature rules to 16 digits of accuracy. The resulting implementation, in the Julia language, is available at <https://github.com/rrhiemstra/quadrules>. It proved to be robust and accurate in computing quadrature rules for non-uniform target spline spaces  $\mathcal{S}_r^q(\Omega)$  of varying polynomial degree  $q$ , regularity  $r$  of the univariate parameterization  $\Omega$  and dimension of the space.

Explicit rules were presented in 16 digits of accuracy for some of the more commonly used uniform spline spaces defined on an open knot vector. These are made available for download at <https://github.com/rrhiemstra/quadrules>. This includes the rule for the target space  $\mathcal{S}_1^5$ , which was previously presented in [7], and the rule corresponding to  $\mathcal{S}_1^6$ . Both these rules have practical value. The latter because it provides full integration of the mass and stiffness matrix in cubic, maximally smooth, isogeometric analysis, for second order operators, such as arise in potential theory and elasticity. The former can server as an inexpensive alternative with slight under-integration.

A major focus of our work, distinguishing our contribution from others, is the application of the computed rules in the context of full and reduced integration applied to two- and three-dimensional boundary value problems involving mapped NURBS geometries and hierarchical refinement. Consider a univariate trial space of polynomial degree  $p$  and  $r$  continuous derivatives. Then a quadrature rule corresponding to the even degree space  $\mathcal{S}_{r-1}^{2p}$  integrates exactly, to 16 significant digits, the elements of the mass and stiffness matrix in a reaction-diffusion problem. We further propose to use odd degree quadrature rules corresponding to a target space  $\mathcal{S}_{r-1}^{2p-1}$  in reduced integration. These strategies are easily extended to the multidimensional setting by means of the tensor product. The numerical benchmark problems demonstrated that substantial savings in quadrature points are possible without any loss in accuracy as compared with full Gauss-Legendre quadrature. Full integration using the optimal rules yielded savings up to a factor of five. Reduced integration proved a mechanism through which another factor of two in efficiency can be attained without affecting accuracy.

Special care is warranted, however, when applying the reduced rules in practice. Reduced integration can, in general, lead to rank insufficient, or singular, stiffness matrices. We proved a theorem rigorously establishing the stability and accuracy of the reduced rules applied to elliptic problems with sufficiently smooth data. In particular, we showed that the error due to reduced integration is  $\mathcal{O}(h^{p+1})$ , and is therefore of higher order than the

discretization error, which is  $\mathcal{O}(h^p)$ ;  $h$  being a measure of the characteristic element size and  $p$  the polynomial degree of the trial space. The stability and accuracy properties of the reduced rules were further investigated using discrete spectrum analysis. Comparison of the eigenvalues with the analytical solution clearly demonstrated their accuracy. Furthermore, no spurious eigenvalues were observed. While not a proof, this study gave us insight into the numerical approximation made over the whole range of modes and has provided us the confidence to propose these rules for use in similar elliptic problems, including ones involving more complicated geometry.

We see several directions for future research. The proof of stability and accuracy of reduced integration did not extend to hierarchically refined splines. While full quadrature has been shown to work well here, reduced integration may lead to a loss of discrete stability. This warrants further investigation. Reduced integration corresponding to a target space of  $\mathcal{S}_{r-1}^{2p-2}$  could also be explored. According to the presented theorem these rules are stable and accurate to  $\mathcal{O}(h^p)$  in the  $H^1$ -norm. Hence, the error due to approximate integration is of the same order as the discretization error. It would be interesting to study the efficacy of these rules applied to NURBS based isogeometric analysis. We would further like to extend optimal reduced and full quadrature strategies to single level discretization techniques with local refinement, such as T-splines and LR B-splines. Finally, in addition to the methodology extending [29] presented herein, there are at least two other approaches with the same objective that have recently appeared. It would be worthwhile to perform a systematic comparison of them.

## References

- [1] C. Adam, S. Bouabdallah, M. Zarroug, and H. Maitournam. Improved numerical integration for locking treatment in isogeometric structural elements, part I: Beams. *Computer Methods in Applied Mechanics and Engineering*, 279:1–28, 2014.
- [2] C. Adam, S. Bouabdallah, M. Zarroug, and H. Maitournam. Improved numerical integration for locking treatment in isogeometric structural elements. part II: Plates and shells. *Computer Methods in Applied Mechanics and Engineering*, 284:106–137, 2015.
- [3] C. Adam, T. J. R. Hughes, S. Bouabdallah, M. Zarroug, and H. Maitournam. Selective and reduced numerical integrations for NURBS-based isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 284:732–761, 2015.
- [4] R. Ait-Haddou, M. Bartoň, and V. M. Calo. Explicit Gaussian quadrature rules for cubic splines with non-uniform knot sequences. *arXiv preprint arXiv:1410.7196*, 2014.
- [5] P. Antolin, A. Buffa, F. Calabrò, M. Martinelli, and G. Sangalli. Efficient matrix computation for tensor-product isogeometric analysis: The use of sum factorization. *Computer Methods in Applied Mechanics and Engineering*, 285:817–828, 2015.
- [6] F. Auricchio, F. Calabrò, T. Hughes, A. Reali, and G. Sangalli. A simple algorithm for obtaining nearly optimal quadrature rules for NURBS-based isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 249:15–27, 2012.
- [7] M. Bartoň, R. Ait-Haddou, and V. M. Calo. Gaussian quadrature rules for  $C^1$  quintic splines. *arXiv preprint arXiv:1503.00907*, 2015.
- [8] M. Bartoň and V. Calo. Optimal quadrature rules for isogeometric analysis. *arXiv preprint arXiv:1511.03882*, 2015.
- [9] M. Bartoň and V. M. Calo. Gaussian quadrature for splines via homotopy continuation: Rules for  $C^2$  cubic splines. *Journal of Computational and Applied Mathematics*, 296:709–723, 2016.
- [10] Y. Bazilevs, L. Beirão da Veiga, J. A. Cottrell, T. J. R. Hughes, and G. Sangalli. Isogeometric analysis: approximation, stability and error estimates for h-refined meshes. *Mathematical Models and Methods in Applied Sciences*, 16(07):1031–1090, 2006.
- [11] L. Beirão da Veiga, A. Buffa, J. Rivas, and G. Sangalli. Some estimates for h-p-k-refinement in isogeometric analysis. *Numerische Mathematik*, 118(2):271–305, 2011.
- [12] P. Bornemann and F. Cirak. A subdivision-based implementation of the hierarchical B-spline finite element method. *Computer Methods in Applied Mechanics and Engineering*, 253:584–598, 2013.
- [13] J. Bremer, Z. Gimbutas, and V. Rokhlin. A nonlinear optimization procedure for generalized Gaussian quadratures. *SIAM Journal on Scientific Computing*, 32(4):1761–1788, 2010.

- [14] F. Calabrò and C. Manni. The choice of quadrature in NURBS-based isogeometric analysis. In *Proc. of the 3rd South-East European Conference on Computational Mechanics (SEECCM)*, 2013.
- [15] H. Cheng, V. Rokhlin, and N. Yarvin. Nonlinear optimization, quadrature, and interpolation. *SIAM Journal on Optimization*, 9(4):901–923, 1999.
- [16] P. G. Ciarlet. *The finite element method for elliptic problems*, volume 40. Siam, 2002.
- [17] J. A. Cottrell, T. J. R. Hughes, and Y. Bazilevs. *Isogeometric analysis: toward integration of CAD and FEA*. John Wiley & Sons, 2009.
- [18] J. A. Cottrell, T. J. R. Hughes, and A. Reali. Studies of refinement and continuity in isogeometric structural analysis. *Computer methods in applied mechanics and engineering*, 196(41):4160–4183, 2007.
- [19] J. A. Cottrell, A. Reali, Y. Bazilevs, and T. J. R. Hughes. Isogeometric analysis of structural vibrations. *Computer methods in applied mechanics and engineering*, 195(41):5257–5296, 2006.
- [20] C. De Boor. A practical guide to splines. *Mathematics of Computation*, 1978.
- [21] J. A. Evans and T. J. R. Hughes. Discrete spectrum analyses for various mixed discretizations of the Stokes eigenproblem. *Computational Mechanics*, 50(6):667–674, 2012.
- [22] C. Giannelli, B. Jüttler, and H. Speleers. THB-splines: The truncated basis for hierarchical splines. *Computer Aided Geometric Design*, 29(7):485–498, 2012.
- [23] G. H. Golub and C. F. Van Loan. *Matrix computations*, volume 3. JHU Press, 2012.
- [24] M. Hillman, J. Chen, and Y. Bazilevs. Variationally consistent domain integration for isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 284:521–540, 2015.
- [25] T. J. R. Hughes. *The finite element method: linear static and dynamic finite element analysis*. Courier Corporation, 2012.
- [26] T. J. R. Hughes, J. A. Cottrell, and Y. Bazilevs. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer methods in applied mechanics and engineering*, 194(39):4135–4195, 2005.
- [27] T. J. R. Hughes, J. A. Evans, and A. Reali. Finite element and NURBS approximations of eigenvalue, boundary-value, and initial-value problems. *Computer Methods in Applied Mechanics and Engineering*, 272:290–320, 2014.
- [28] T. J. R. Hughes, A. Reali, and G. Sangalli. Duality and unified analysis of discrete approximations in structural dynamics and wave propagation: comparison of p-method finite elements with k-method NURBS. *Computer methods in applied mechanics and engineering*, 197(49):4104–4124, 2008.

- [29] T. J. R. Hughes, A. Reali, and G. Sangalli. Efficient quadrature for NURBS-based isogeometric analysis. *Computer methods in applied mechanics and engineering*, 199(5):301–313, 2010.
- [30] K. A. Johannessen. Optimal quadrature rules for univariate and tensor product splines. *Submitted to Computational Methods In Applied Mechanics*, 2016.
- [31] A. Karatarakis, P. Karakitsios, and M. Papadrakakis. GPU accelerated computation of the isogeometric analysis stiffness matrix. *Computer Methods in Applied Mechanics and Engineering*, 269:334–355, 2014.
- [32] S. Karlin and C. Micchelli. The fundamental theorem of algebra for monosplines satisfying boundary conditions. *Israel Journal of Mathematics*, 11(4):405–451, 1972.
- [33] S. Karlin and W. J. Studden. *Tchebycheff systems: With applications in analysis and statistics*, volume 376. Interscience Publishers New York, 1966.
- [34] R. Kraft. *Adaptive and linearly independent multilevel B-splines*. SFB 404, Geschäftsstelle, 1997.
- [35] S. Lipton, J. A. Evans, Y. Bazilevs, T. Elguedj, and T. J. R. Hughes. Robustness of isogeometric structural discretizations under severe mesh distortion. *Computer Methods in Applied Mechanics and Engineering*, 199(5):357–373, 2010.
- [36] J. Ma, V. Rokhlin, and S. Wandzura. Generalized Gaussian quadrature rules for systems of arbitrary functions. *SIAM Journal on Numerical Analysis*, 33(3):971–996, 1996.
- [37] A. Mantzaflaris and B. Jüttler. Exploring matrix generation strategies in isogeometric analysis. In *Mathematical Methods for Curves and Surfaces*, pages 364–382. Springer, 2014.
- [38] A. Mantzaflaris and B. Jüttler. Integration by interpolation and look-up for Galerkin-based isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 284:373–400, 2015.
- [39] C. Micchelli and A. Pinkus. Moment theory for weak chebyshev systems with applications to monosplines, quadrature formulae and best one-sided  $L^1$ -approximation by spline functions with fixed knots. *SIAM Journal on Mathematical Analysis*, 8(2):206–230, 1977.
- [40] G. Nikolov. On certain definite quadrature formulae. *Journal of computational and applied mathematics*, 75(2):329–343, 1996.
- [41] D. Rypl and B. Patzák. Study of computational efficiency of numerical quadrature schemes in the isogeometric analysis. *Engineering Mechanics*, page 304, 2012.
- [42] D. Schillinger, L. Dede, M. A. Scott, J. A. Evans, M. J. Borden, E. Rank, and T. J. R. Hughes. An isogeometric design-through-analysis methodology based on adaptive hierarchical refinement of NURBS, immersed boundary methods, and T-spline CAD surfaces. *Computer Methods in Applied Mechanics and Engineering*, 249:116–150, 2012.

- [43] D. Schillinger, J. A. Evans, A. Reali, M. A. Scott, and T. J. R. Hughes. Isogeometric collocation: Cost comparison with Galerkin methods and extension to adaptive hierarchical NURBS discretizations. *Computer Methods in Applied Mechanics and Engineering*, 267:170–232, 2013.
- [44] D. Schillinger, S. J. Hossain, and T. J. R. Hughes. Reduced Bézier element quadrature rules for quadratic and cubic splines in isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 277:1–45, 2014.
- [45] I. Schoenberg. Spline functions, convex curves and mechanical quadrature. *Bulletin of the American Mathematical Society*, 64(6):352–357, 1958.
- [46] L. Schumaker. *Spline functions: basic theory*. Cambridge University Press, 2007.
- [47] G. Strang and G. J. Fix. *An analysis of the finite element method*, volume 212. Prentice-Hall Englewood Cliffs, NJ, 1973.
- [48] A. H. Stroud and D. Secrest. *Gaussian quadrature formulas*, volume 39. Prentice-Hall Englewood Cliffs, NJ, 1966.
- [49] L. N. Trefethen and D. Bau III. *Numerical linear algebra*, volume 50. Siam, 1997.
- [50] A.-V. Vuong, C. Giannelli, B. Jüttler, and B. Simeon. A hierarchical approach to adaptive local refinement in isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 200(49):3554–3567, 2011.