



# Symmetric quadrature rules for simplexes based on sphere close packed lattice arrangements



D.M. Williams<sup>a,\*</sup>, L. Shunn<sup>b</sup>, A. Jameson<sup>a</sup>

<sup>a</sup> Department of Aeronautics and Astronautics, Stanford University, Stanford, CA 94305, USA

<sup>b</sup> Cascade Technologies, Palo Alto, CA 94303, USA

## HIGHLIGHTS

- A general approach for computing optimal symmetric quadrature rules on simplexes is identified.
- A new family of optimal symmetric quadrature rules on the 2-simplex (triangle) is identified.
- A new optimal symmetric quadrature rule on the 3-simplex (tetrahedron) is identified.

## ARTICLE INFO

### Article history:

Received 11 July 2013

Received in revised form 6 January 2014

### Keywords:

Quadrature  
Integration  
Symmetric  
Triangle  
Tetrahedron  
Simplex

## ABSTRACT

Sphere close packed (SCP) lattice arrangements of points are well-suited for formulating symmetric quadrature rules on simplexes, as they are symmetric under affine transformations of the simplex unto itself in 2D and 3D. As a result, SCP lattice arrangements have been utilized to formulate symmetric quadrature rules with  $N_p = 1, 4, 10, 20, 35$ , and 56 points on the 3-simplex (Shunn and Ham, 2012). In what follows, the work on the 3-simplex is extended, and SCP lattices are employed to identify symmetric quadrature rules with  $N_p = 1, 3, 6, 10, 15, 21, 28, 36, 45, 55$ , and 66 points on the 2-simplex and  $N_p = 84$  points on the 3-simplex. These rules are found to be capable of exactly integrating polynomials of up to degree 17 in 2D and up to degree 9 in 3D.

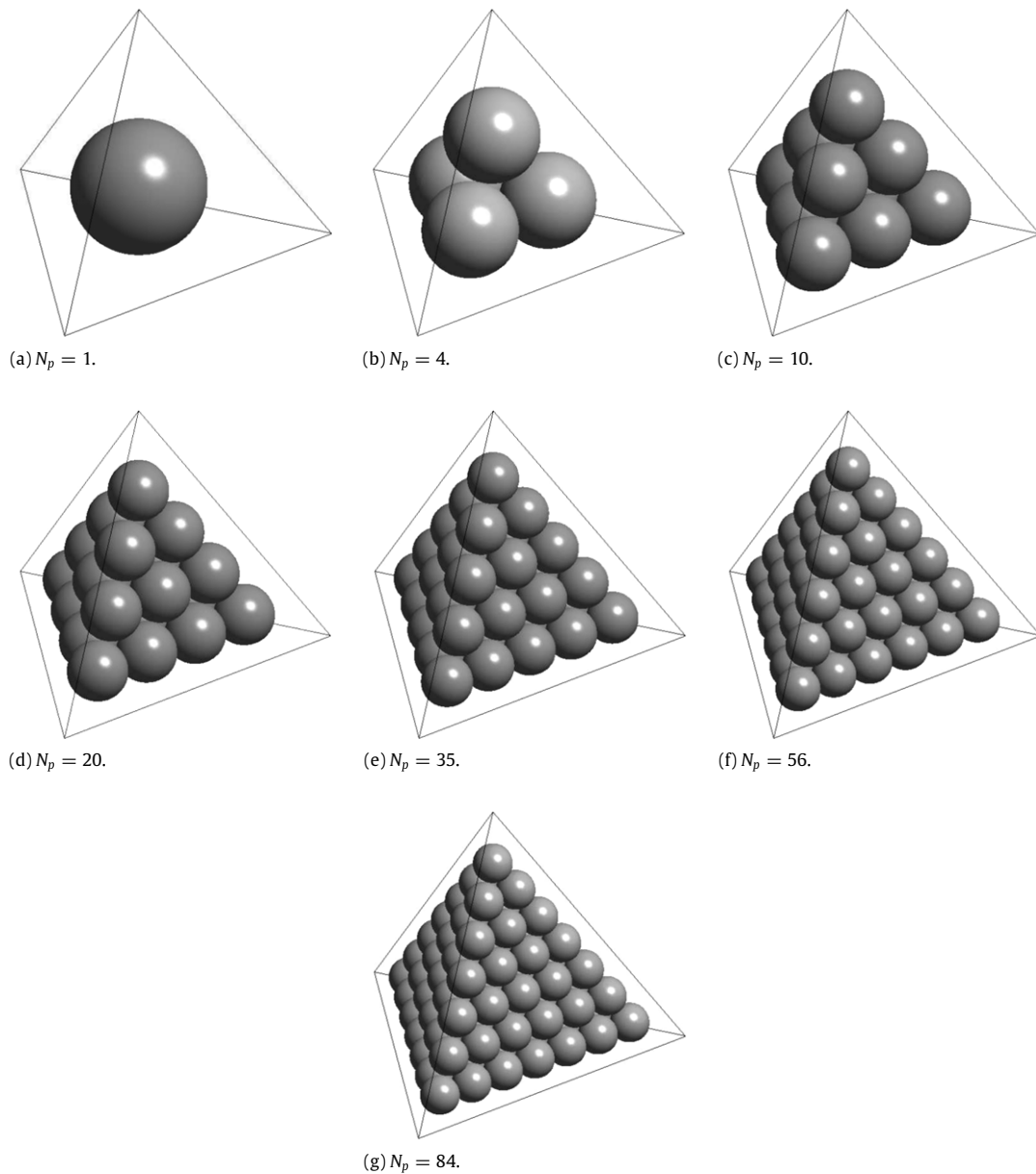
© 2014 Elsevier B.V. All rights reserved.

## 1. Introduction

There have been significant efforts to identify high-order quadrature rules on  $d$ -simplex and  $d$ -hypercube geometries as evidenced by the surveys in [1–5]. On the  $d$ -hypercube, it is well-known that optimal (or near optimal) quadrature rules with  $N_p = n^d$  points can be constructed from tensor products of 1D Gauss–Legendre quadrature rules with  $n$  points [6]. These quadrature rules are widely used because, in addition to their optimality (or near optimality), they are symmetric (as they are invariant under reflections and rotations that map the  $d$ -hypercube unto itself), possess positive weights, and have points located within the interior of the  $d$ -hypercube. Due to these desirable properties, there have been attempts to extend the Gauss–Legendre rules to  $d$ -simplexes (cf. [6,7]). The simplest of such approaches has involved constructing Gauss–Legendre rules on the  $d$ -hypercube and then degenerating vertices until the  $d$ -simplex is obtained. However, in general the resulting rules are no longer symmetric on the  $d$ -simplex, as they contain anisotropic clusters of points near the degenerate vertices. In addition, the optimality of these rules has yet to be shown analytically. In fact, to the authors' knowledge, no one has identified a family of symmetric quadrature rules on the  $d$ -simplex for which optimality can be rigorously proven. For this reason, the formulation of quadrature rules on the  $d$ -simplex remains an open area of research, as demonstrated by the recent work presented in [8–17].

\* Corresponding author. Tel.: +1 248 721 3945; fax: +1 650 723 3018.

E-mail addresses: [davidmw@alumni.stanford.edu](mailto:davidmw@alumni.stanford.edu) (D.M. Williams), [lee@shunn.net](mailto:lee@shunn.net) (L. Shunn), [jameson@baboon.stanford.edu](mailto:jameson@baboon.stanford.edu) (A. Jameson).



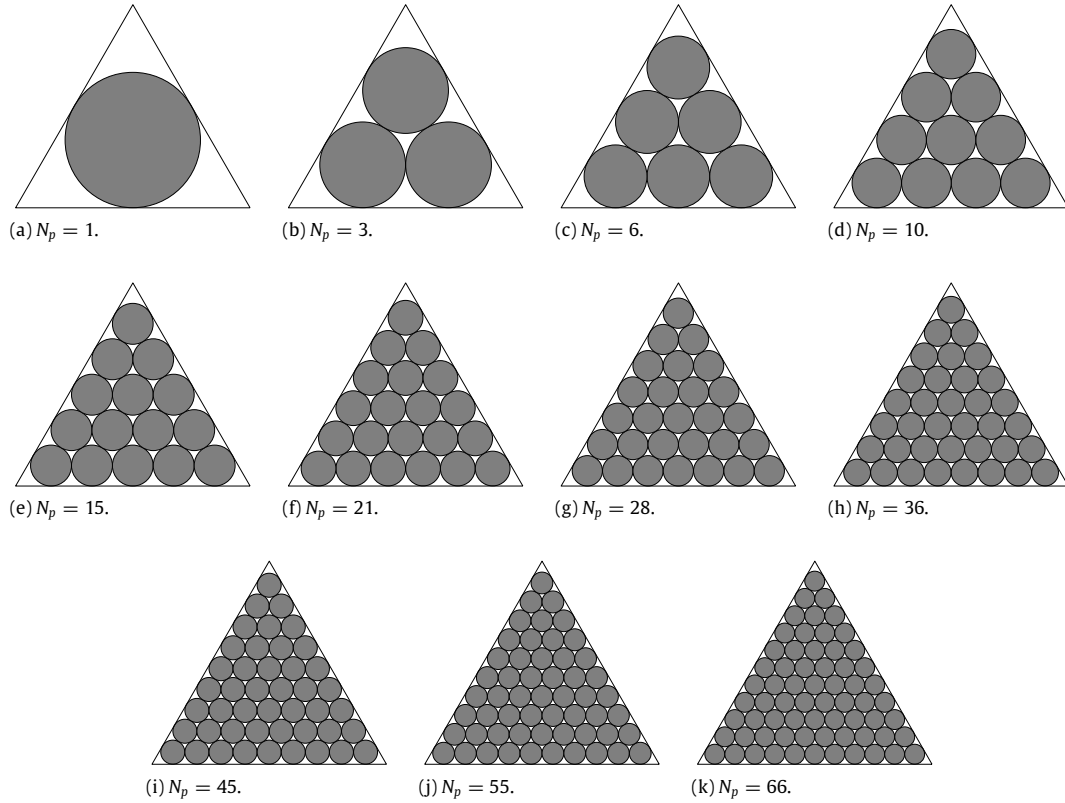
**Fig. 1.** Cubic close packed (CCP) configurations on tetrahedra with  $N_p = 1, 4, 10, 20, 35, 56$ , and  $84$  points.

Of particular interest, is the effort by Shunn and Ham in [17] to construct quadrature rules on the 3-simplex based on cubic close packing (CCP) arrangements of points. In their work, the CCP arrangements of points (i.e., the CCP lattices) are defined by the centers of spheres in the CCP configuration, as shown in Fig. 1 for the cases of  $N_p = 1, 4, 10, 20, 35, 56$ , and  $84$ .

The CCP lattices are viewed as the 3-simplex analog to the uniform cartesian lattices on which the Gauss–Legendre rules on the 3-hypercube are based, in the sense that they possess the same properties of symmetry on the 3-simplex as uniform cartesian lattices possess on the 3-hypercube. Based on this fact, it was supposed that an optimal family of symmetric quadrature rules on the 3-simplex could be obtained by employing the CCP lattices as initial conditions to optimization procedures for identifying the rules. Following this approach, a family of symmetric, locally optimal quadrature rules on the 3-simplex with  $N_p = 1, 4, 10, 20, 35$ , and  $56$  points was obtained [17].

This work attempts to extend the approach in [17] to identify new quadrature rules on the  $d$ -simplex for the cases of  $d = 2$  and  $d = 3$ . This extension requires the construction of CCP lattices in 2-space (sometimes referred to as ‘hexagonal packing lattices’), which are shown in Fig. 2 for the cases of  $N_p = 1, 3, 6, 10, 15, 21, 28, 36, 45, 55$ , and  $66$ .

For convenience, the analogs of the CCP lattices in  $d$ -space will henceforth be referred to as  $d$ -sphere close packed lattices ( $d$ -SCP lattices). It is useful to note that, in general, the number of points in the  $d$ -SCP lattices (the values of  $N_p$  for the lattices)



**Fig. 2.** Cubic close packed (CCP) configurations on triangles with  $N_p = 1, 3, 6, 10, 15, 21, 28, 36, 45, 55$ , and  $66$  points.

can be expressed as a function of  $N_l$ , the number of layers of  $d$ -spheres in the lattices, as follows

$$N_p = \frac{(N_l - 1 + d)!}{d! (N_l - 1)!}, \quad (1)$$

where each layer is of dimension  $d - 1$ .

The remainder of this article is structured as follows. Section 2 presents requirements for the quadrature rules and introduces the theoretical machinery that enables these requirements to be enforced. Section 3 discusses the practical procedure for obtaining the quadrature rules through solving nonlinearly constrained optimization problems. Section 4 and Appendices A and B summarize the quadrature point locations and weights for the resulting rules. Finally, Section 5 presents the results of numerical experiments evaluating the orders of accuracy and absolute errors of the quadrature rules.

## 2. Overview of the theory of quadrature rules on simplexes

This section will describe the theoretical basis of the standard approach for finding quadrature rules on  $d$ -simplexes. The description presented here is a generalization of the description of Shunn and Ham in [17].

### 2.1. Preliminary definition of the problem

Consider the domain  $\Omega$  which resides in a  $d$ -dimensional space with coordinate vector  $\mathbf{x} \in \mathbb{R}^d$ . Suppose that a scalar-valued function  $f(\mathbf{x})$  is well-defined on  $\Omega$  (such that  $f(\mathbf{x}) \in \mathbb{R} \forall \mathbf{x} \in \Omega$ ) and that the integral of  $f(\mathbf{x})$  on  $\Omega$  is also well-defined such that

$$F \equiv \int_{\Omega} f(\mathbf{x}) d\Omega, \quad F \in \mathbb{R}. \quad (2)$$

In this case,  $F$  can be approximated using a quadrature rule ( $\mathcal{F}$ ) as follows

$$\mathcal{F} \equiv V(\Omega) \left[ \sum_{i=1}^{N_p} w_i f(\mathbf{x}_i) \right] \approx F, \quad (3)$$

where  $V(\Omega)$  denotes the volume of  $\Omega$ , and  $\mathbf{x}_i$  and  $w_i$  denote the locations and weights of the quadrature points, respectively. Note that the error  $\epsilon$  introduced by the quadrature approximation in Eq. (3) can be defined as follows

$$\epsilon \equiv \frac{F - \mathcal{F}}{V(\Omega)}. \quad (4)$$

Now, suppose that  $\Omega$  is defined to be the  $d$ -simplex with vertices  $\mathbf{v}_1, \dots, \mathbf{v}_{d+1}$ . One may redefine the quadrature rule in Eq. (3) in terms of the  $d$ -simplicial definitions of the volume  $V(\Omega)$  and the quadrature point locations  $\mathbf{x}_i$ . In particular, the volume of the  $d$ -simplex can be defined using the Cayley–Menger determinant as follows

$$V(\Omega) = \frac{1}{d!} \det(\mathbf{B}), \quad (5)$$

where  $\mathbf{B}$  is a matrix which has columns that are constructed from the differences between vertices of the  $d$ -simplex, i.e.,

$$\mathbf{B} = \{(\mathbf{v}_2 - \mathbf{v}_1) \cdots (\mathbf{v}_{d+1} - \mathbf{v}_1)\}. \quad (6)$$

In addition, the quadrature point locations  $\mathbf{x}_i$  can be expressed as the following linear combinations of the  $d$ -simplex vertices

$$\mathbf{x}_i = \sum_{j=1}^{d+1} a_{i,j} \mathbf{v}_j. \quad (7)$$

Upon substituting the expressions for  $V(\Omega)$  and  $\mathbf{x}_i$  from Eqs. (5) and (7) into Eq. (3), one obtains the following general expression for the quadrature rule  $\mathcal{F}$  on the  $d$ -simplex

$$\mathcal{F} = \frac{1}{d!} \det(\mathbf{B}) \left[ \sum_{i=1}^{N_p} w_i f \left( \sum_{j=1}^{d+1} a_{i,j} \mathbf{v}_j \right) \right]. \quad (8)$$

## 2.2. Quadrature rule requirements

Before proceeding further, it is useful to review the requirements that quadrature rules are usually constrained to satisfy. In accordance with the discussions in [3], each quadrature rule is required to:

- Integrate a polynomial of the highest possible order and minimize the magnitude of the truncation error term.
- Ensure that all quadrature weights are non-negative in order to minimize the possibility of cancellation errors.
- Ensure that there are no quadrature points located outside of the  $d$ -simplex.
- Ensure that all quadrature points are symmetrically arranged within the  $d$ -simplex. The quadrature rule must be invariant under affine transformations of the  $d$ -simplex unto itself.

In addition, on the 2-simplex it is desirable (but not required) that all quadrature rules preserve the layered structure of the corresponding 2-SCP configuration. More precisely, it is convenient if the quadrature rules have rows of points for which the number of points is the same as the number of points in a complementary row of the 2-SCP configuration. Here it is important to note that a row of points is defined such that the  $y$  coordinates of all the points in a given row are greater than all  $y$  coordinates of the points in the row below and less than all  $y$  coordinates of points in the row above (with natural exceptions for the top-most and bottom-most rows). When the points can be partitioned into rows in this way, it facilitates the creation of mappings between the indices of quadrature points on pairs of 2-simplexes. These mappings are useful in certain applications, for example, in numerical integration procedures on triangular (2-simplex) faces of elements in 3D meshes of tetrahedra (3-simplexes) that are frequently employed in discontinuous finite element methods. Refer to [Appendix A](#) for details regarding this topic.

## 2.3. Enforcement of quadrature rule requirements

Evidently, the objective is to find the particular rule (or set of rules) that satisfies the requirements of the previous section. Of these requirements, it is perhaps most important to find a quadrature rule which satisfies the first one, namely finding a quadrature rule which integrates a polynomial of the highest possible degree and minimizes the approximation error  $\epsilon$  (Eq. (4)) for a given number of quadrature points  $N_p$ . More precisely,  $\epsilon \sim O(\delta^n)$  must be minimized, where  $n$  is the highest possible order, and where  $\delta$  is the ‘characteristic’ length of edges between neighboring vertices  $\mathbf{v}_k$  and  $\mathbf{v}_{k+1}$  of the  $d$ -simplex (where  $k \leq d$ ). In order to begin finding a quadrature rule that meets this criterion, consider approximating  $f(\mathbf{x})$  with a  $n$ th order Taylor series expansion about the centroid  $\mathbf{x}_0$  of  $\Omega$  (denoted by  $f_n(\mathbf{x})$ ) where

$$f_n(\mathbf{x}) \equiv f(\mathbf{x}) + O(\delta^n), \quad (9)$$

so that Eq. (8) becomes

$$\mathcal{F} = \frac{1}{d!} \det(\mathbf{B}) \left[ \sum_{i=1}^{N_p} w_i f_n \left( \sum_{j=1}^{d+1} a_{i,j} \mathbf{v}_j \right) \right] + O(\delta^{n+d}). \quad (10)$$

Setting Eq. (10) aside for the moment, consider integrating both sides of Eq. (9) over the domain  $\Omega$  in order to obtain the following expression for  $F$

$$F = \int_{\Omega} f_n(\mathbf{x}) d\Omega + O(\delta^{n+d}) \equiv F_n + O(\delta^{n+d}). \quad (11)$$

Upon substituting  $\mathcal{F}$  and  $F$  from Eqs. (10) and (11), and  $V(\Omega)$  from Eq. (5) into Eq. (4), one obtains the following expression for the error term  $\epsilon$

$$\epsilon = \frac{F_n d!}{\det(\mathbf{B})} - \left[ \sum_{i=1}^{N_p} w_i f_n \left( \sum_{j=1}^{d+1} a_{i,j} \mathbf{v}_j \right) \right] + O(\delta^n), \quad (12)$$

which is equivalent to

$$\epsilon = \epsilon_n + O(\delta^n), \quad (13)$$

where

$$\epsilon_n \equiv \frac{F_n d!}{\det(\mathbf{B})} - \left[ \sum_{i=1}^{N_p} w_i f_n \left( \sum_{j=1}^{d+1} a_{i,j} \mathbf{v}_j \right) \right]. \quad (14)$$

It is well-known that  $\epsilon_n$  is proportional to  $\delta^n$  [17]. Therefore, upon combining this result and the result of Eq. (13), one concludes that

$$\epsilon \sim O(\delta^n), \quad (15)$$

as expected.

From this analysis, it can be seen that the quadrature rule error scales with the order of the term  $\epsilon_n$ . Therefore, satisfying the first quadrature rule requirement simplifies to optimizing the quadrature point locations and weights such that  $\epsilon_n$  is minimized for the maximum possible value of  $n$ .

The remaining quadrature rule requirements can be satisfied as follows. The second requirement states that the quadrature weights must be non-negative, which is equivalent to requiring that

$$w_i \geq 0. \quad (16)$$

In addition, the third requirement states that the quadrature points must not lie outside of the  $d$ -simplex, which is equivalent to placing the following conditions on the coefficients  $a_{i,j}$

$$a_{i,j} \geq 0, \quad \sum_{j=1}^{d+1} a_{i,j} = 1. \quad (17)$$

The fourth requirement states that the quadrature points must be arranged in a symmetric fashion within the  $d$ -simplex. In order to satisfy this requirement, one must enforce additional constraints on the coefficients  $a_{i,j}$ . Prior to formulating these constraints, it is convenient to first formulate a symmetric  $d$ -simplex on which to enforce them. Of course, symmetry constraints may be enforced on any  $d$ -simplex, as a linear mapping to transform quadrature points from the initial simplex to any other simplex of the same dimension can always be defined. However, it is more convenient to construct these constraints on the standard, equilateral  $d$ -simplex (denoted by  $\Omega_S$ ) whose centroid ( $\mathbf{x}_0$ ) is located at the origin. Evidently,  $\Omega_S$  is convenient to utilize because it has  $d!$ -fold symmetry in the sense that all of the  $d!$  unique, affine, symmetric transformations that are possible for the  $d$ -simplex merely result in  $\Omega_S$  being transformed unto itself. However, it also possesses some additional useful properties. In particular, the 'characteristic' edge length  $\delta$  is straightforward to define, as the length of each edge is identical. Furthermore, because  $\mathbf{x}_0$  is located at the origin, one is ensured that Taylor series expansions about  $\mathbf{x}_0$  are simple to formulate, as all terms which contain components of the vector  $(\mathbf{x} - \mathbf{x}_0)^m$  can be reduced to contain only components of  $\mathbf{x}^m$ .

Now, having established the useful properties of the equilateral  $d$ -simplex  $\Omega_S$ , it is important to examine the analytical form for the equilateral 2-simplex and 3-simplex that will serve as the focus for the remainder of this article. The equilateral 2-simplex with edge length  $\delta$  is formed from the following three vertices

$$\begin{aligned} \mathbf{v}_1 &= \delta \left( -\frac{1}{2}, -\frac{\sqrt{3}}{6} \right) \\ \mathbf{v}_2 &= \delta \left( \frac{1}{2}, -\frac{\sqrt{3}}{6} \right) \\ \mathbf{v}_3 &= \delta \left( 0, \frac{\sqrt{3}}{3} \right). \end{aligned} \quad (18)$$

Upon substituting the vertices from Eq. (18) into Eq. (5), one finds that the 2-simplex has a volume of  $V(\Omega_S) = \delta^2 \sqrt{3}/4$ . In addition, it is useful to note that Eq. (2) can be reformulated on the 2-simplex as follows

$$F = \int_{y_0}^{y_1} \int_{x_0}^{x_1} f(x, y) dx dy, \quad (19)$$

where the limits of integration  $x_0, x_1, y_0$ , and  $y_1$  are defined as follows

$$x_0 = -\frac{\sqrt{3}}{3} \left( \frac{\sqrt{3}}{3} \delta - y \right) \quad x_1 = \frac{\sqrt{3}}{3} \left( \frac{\sqrt{3}}{3} \delta - y \right) \quad (20)$$

$$y_0 = -\frac{\sqrt{3}}{6} \delta \quad y_1 = \frac{\sqrt{3}}{3} \delta. \quad (21)$$

The equilateral 3-simplex with edge length  $\delta$  is formed from the following four vertices

$$\begin{aligned} \mathbf{v}_1 &= \delta \left( -\frac{1}{2}, -\frac{\sqrt{3}}{6}, -\frac{\sqrt{6}}{12} \right) \\ \mathbf{v}_2 &= \delta \left( \frac{1}{2}, -\frac{\sqrt{3}}{6}, -\frac{\sqrt{6}}{12} \right) \\ \mathbf{v}_3 &= \delta \left( 0, \frac{\sqrt{3}}{3}, -\frac{\sqrt{6}}{12} \right) \\ \mathbf{v}_4 &= \delta \left( 0, 0, \frac{\sqrt{6}}{4} \right). \end{aligned} \quad (22)$$

Upon substituting the vertices from Eq. (22) into Eq. (5), one finds that the 3-simplex has a volume of  $V(\Omega_S) = \delta^3 \sqrt{2}/12$ . In addition, it is useful to note that Eq. (2) can be reformulated on the 3-simplex as follows

$$F = \int_{z_0}^{z_1} \int_{y_0}^{y_1} \int_{x_0}^{x_1} f(x, y) dx dy dz, \quad (23)$$

where the limits of integration  $x_0, x_1, y_0, y_1, z_0$ , and  $z_1$  are defined as follows

$$x_0 = -\frac{\sqrt{3}}{3} \left[ \frac{\sqrt{2}}{2} \left( \frac{\sqrt{6}}{4} \delta - z \right) - y \right] \quad x_1 = \frac{\sqrt{3}}{3} \left[ \frac{\sqrt{2}}{2} \left( \frac{\sqrt{6}}{4} \delta - z \right) - y \right] \quad (24)$$

$$y_0 = -\frac{\sqrt{2}}{4} \left( \frac{\sqrt{6}}{4} \delta - z \right) \quad y_1 = \frac{\sqrt{2}}{2} \left( \frac{\sqrt{6}}{4} \delta - z \right) \quad (25)$$

$$z_0 = -\frac{\sqrt{6}}{12} \delta \quad z_1 = \frac{\sqrt{6}}{4} \delta. \quad (26)$$

Having established precise definitions for the  $d = 2$  and  $d = 3$  standard simplexes, symmetry constraints on the coefficients  $a_{i,j}$  for each of these simplexes may now be formulated. For convenience, one may impose the constraints by introducing parameters  $\alpha_k$  which symmetrically parameterize the point locations  $\mathbf{x}_i$ , and indirectly via Eq. (7) parameterize (and constrain) the  $a_{i,j}$ 's. In this way, the problem simplifies to finding the symmetry parameters  $\alpha_k$ , from which  $\mathbf{x}_i$  ( $\alpha_k$ ) and  $a_{i,j}$  ( $\mathbf{x}_i$ ) immediately follow. This is convenient because there are frequently far fewer parameters  $\alpha_k$  than there are coefficients  $a_{i,j}$ . For example, consider the symmetric parameterization of the 2-simplex with  $N_p = 15$  points (as illustrated in Fig. 2(e)). In this case, there are a total of  $15(d+1) = 45$  unknown coefficients  $a_{i,j}$ , whereas there are only five unknown parameters  $\alpha_k$  (where  $k = 1, \dots, 5$ ). The reduction in the number of variables arises naturally from the symmetry requirements. In particular, due to symmetry, six of the fifteen points are required to lie along lines between the centroid  $\mathbf{x}_0$  and the vertices  $\mathbf{v}_1, \mathbf{v}_2$ , and  $\mathbf{v}_3$  of the 2-simplex. Because there are three vertices, there are a total of two points along lines between each vertex and the centroid, and there are in turn two symmetry parameters  $\alpha_1$  and  $\alpha_2$  which define the point locations  $\mathbf{x}_i$  as follows

$$\begin{aligned} \mathbf{x}_i &= \alpha_1 \mathbf{v}_i \quad \text{where } i = 1, 2, 3 \\ \mathbf{x}_i &= \alpha_2 \mathbf{v}_{i-3} \quad \text{where } i = 4, 5, 6. \end{aligned} \quad (27)$$

Furthermore, there are three points which are required to lie along lines between the centroid and the midpoints of the simplex edges. The locations of these points can be defined using a single symmetry parameter  $\alpha_3$  as follows

$$\mathbf{x}_i = \alpha_3 (\mathbf{v}_j + \mathbf{v}_m) \quad \text{where } i = 7, 8, 9, \quad 1 \leq j, m \leq 3, \quad j \neq m. \quad (28)$$

Finally, there are six points which must lie along lines which emanate from the centroid and intersect the simplex edges at locations between the edge midpoints and the vertices. The locations of these points can be defined using two symmetry parameters  $\alpha_4$  and  $\alpha_5$  as follows

$$\begin{aligned} \mathbf{x}_i &= \alpha_4 \mathbf{v}_j + \alpha_5 \mathbf{v}_m & \text{where } i = 10, 11, 12 \\ \mathbf{x}_i &= \alpha_5 \mathbf{v}_j + \alpha_4 \mathbf{v}_m & \text{where } i = 13, 14, 15. \end{aligned} \quad (29)$$

Eqs. (27)–(29) provide a complete description of the symmetric point locations in terms of the five symmetry variables  $\alpha_1, \dots, \alpha_5$ .

Next, in order to recover the coefficients  $a_{i,j}$  from the parameters  $\alpha_k$ , one may reformulate Eq. (7) in terms of the following linear system

$$\left\{ \begin{pmatrix} \mathbf{v}_1 \\ 1 \end{pmatrix} \begin{pmatrix} \mathbf{v}_2 \\ 1 \end{pmatrix} \begin{pmatrix} \mathbf{v}_3 \\ 1 \end{pmatrix} \right\} \begin{Bmatrix} a_{i,1} \\ a_{i,2} \\ a_{i,3} \end{Bmatrix} = \begin{Bmatrix} x_i \\ y_i \\ 1 \end{Bmatrix}, \quad (30)$$

where the fact that  $a_{i,1} + a_{i,2} + a_{i,3} = 1$  (Eq. (17)) has been used.

A similar procedure for enforcing the symmetry constraints can be employed on the 3-simplex. For example, consider the symmetric arrangement of  $N_p = 84$  points on the 3-simplex (as illustrated in Fig. 1(g)). There are a total of  $84(d+1) = 336$  parameters  $a_{i,j}$ , however, it turns out that symmetry constraints need only be enforced on fourteen parameters  $\alpha_k$  (where  $k = 1, \dots, 14$ ). In particular, due to symmetry, eight of the eighty-four points must be located along lines from the centroid  $\mathbf{x}_0$  to each of the vertices  $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ , and  $\mathbf{v}_4$ . The locations of these points can be expressed in terms of the symmetry parameters  $\alpha_1$  and  $\alpha_2$  as follows

$$\begin{aligned} \mathbf{x}_i &= \alpha_1 \mathbf{v}_i & \text{where } i = 1, 2, 3, 4 \\ \mathbf{x}_i &= \alpha_2 \mathbf{v}_{i-4} & \text{where } i = 5, 6, 7, 8. \end{aligned} \quad (31)$$

In addition, twelve points must be located along lines between the centroid and the midpoints of the simplex edges. The locations of these points can be expressed in terms of the symmetry parameters  $\alpha_3$  and  $\alpha_4$  as follows

$$\begin{aligned} \mathbf{x}_i &= \alpha_3 (\mathbf{v}_j + \mathbf{v}_m) & \text{where } i = 9, \dots, 14 \quad 1 \leq j, m \leq 4, j \neq m \\ \mathbf{x}_i &= \alpha_4 (\mathbf{v}_j + \mathbf{v}_m) & \text{where } i = 15, \dots, 20. \end{aligned} \quad (32)$$

Furthermore, twenty-four points must be located along lines which emanate from the centroid and intersect the simplex edges at locations between the edge midpoints and the vertices. The locations of these points can be expressed in terms of the symmetry parameters  $\alpha_5, \alpha_6, \alpha_7$ , and  $\alpha_8$  as follows

$$\begin{aligned} \mathbf{x}_i &= \alpha_5 \mathbf{v}_j + \alpha_6 \mathbf{v}_m & \text{where } i = 21, \dots, 26 \\ \mathbf{x}_i &= \alpha_6 \mathbf{v}_j + \alpha_5 \mathbf{v}_m & \text{where } i = 27, \dots, 32 \\ \mathbf{x}_i &= \alpha_7 \mathbf{v}_j + \alpha_8 \mathbf{v}_m & \text{where } i = 33, \dots, 38 \\ \mathbf{x}_i &= \alpha_8 \mathbf{v}_j + \alpha_7 \mathbf{v}_m & \text{where } i = 39, \dots, 44. \end{aligned} \quad (33)$$

Finally, the locations of the remaining forty points are composed from all possible unique, symmetric, linear combinations of the three vertices of each face. The locations of these points can be expressed in terms of the symmetry parameters  $\alpha_9, \dots, \alpha_{14}$  as follows

$$\begin{aligned} \mathbf{x}_i &= \alpha_9 (\mathbf{v}_j + \mathbf{v}_m + \mathbf{v}_l) & \text{where } i = 45, 46, 47, 48 \\ & & 1 \leq j, m, l \leq 4, j \neq m \neq l \\ \mathbf{x}_i &= \alpha_{10} \mathbf{v}_j + \alpha_{11} (\mathbf{v}_m + \mathbf{v}_l) & \text{where } i = 49, 50, 51, 52 \\ \mathbf{x}_i &= \alpha_{10} \mathbf{v}_l + \alpha_{11} (\mathbf{v}_j + \mathbf{v}_m) & \text{where } i = 53, 54, 55, 56 \\ \mathbf{x}_i &= \alpha_{10} \mathbf{v}_m + \alpha_{11} (\mathbf{v}_l + \mathbf{v}_j) & \text{where } i = 57, 58, 59, 60 \\ \mathbf{x}_i &= \alpha_{12} \mathbf{v}_j + \alpha_{13} \mathbf{v}_m + \alpha_{14} \mathbf{v}_l & \text{where } i = 61, 62, 63, 64 \\ \mathbf{x}_i &= \alpha_{12} \mathbf{v}_l + \alpha_{13} \mathbf{v}_j + \alpha_{14} \mathbf{v}_m & \text{where } i = 65, 66, 67, 68 \\ \mathbf{x}_i &= \alpha_{12} \mathbf{v}_m + \alpha_{13} \mathbf{v}_l + \alpha_{14} \mathbf{v}_j & \text{where } i = 69, 70, 71, 72 \\ \mathbf{x}_i &= \alpha_{12} \mathbf{v}_j + \alpha_{14} \mathbf{v}_m + \alpha_{13} \mathbf{v}_l & \text{where } i = 73, 74, 75, 76 \\ \mathbf{x}_i &= \alpha_{12} \mathbf{v}_l + \alpha_{14} \mathbf{v}_j + \alpha_{13} \mathbf{v}_m & \text{where } i = 77, 78, 79, 80 \\ \mathbf{x}_i &= \alpha_{12} \mathbf{v}_m + \alpha_{14} \mathbf{v}_l + \alpha_{13} \mathbf{v}_j & \text{where } i = 81, 82, 83, 84. \end{aligned} \quad (34)$$

Eqs. (31)–(34) provide a complete description of the symmetric point locations in terms of the fourteen symmetry variables  $\alpha_1, \dots, \alpha_{14}$ .



One may recover the coefficients  $a_{i,j}$  from the parameters  $\alpha_k$  by solving the following linear system of equations

$$\left\{ \begin{pmatrix} \mathbf{v}_1 \\ 1 \end{pmatrix} \begin{pmatrix} \mathbf{v}_2 \\ 1 \end{pmatrix} \begin{pmatrix} \mathbf{v}_3 \\ 1 \end{pmatrix} \begin{pmatrix} \mathbf{v}_4 \\ 1 \end{pmatrix} \right\} \begin{Bmatrix} a_{i,1} \\ a_{i,2} \\ a_{i,3} \\ a_{i,4} \end{Bmatrix} = \begin{Bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{Bmatrix}, \quad (35)$$

where the fact that  $a_{i,1} + a_{i,2} + a_{i,3} + a_{i,4} = 1$  (Eq. (17)) has been used.

Finally, the fifth requirement (which is a soft requirement) states that the quadrature points on the 2-simplex should be arranged in a way that is consistent with the 2-SCP layering. Since this is not a strict requirement, it need not be enforced directly. However, it can be encouraged by employing 2-SCP configurations (or perturbed 2-SCP configurations) as initial conditions for the optimization process utilized in identifying the quadrature rules. This process will be discussed in more detail in the next section.

### 3. Computation of optimal quadrature rules

Quadrature rules on the 2 and 3-simplexes can be obtained by solving a series of constrained optimization problems. In each of these problems, the symmetry variables ( $\alpha_k$ 's) and the quadrature weights ( $w_i$ 's) are treated as unknowns. These unknowns are subject to the inequality and equality constraints put forth in Section 2.3, which, for convenience, can be reformulated as follows.

#### 3.1. Reformulating the inequality and equality constraints

Inequality constraints on the unknowns are obtained from Eqs. (16) and (17). The constraints in Eq. (16) apply directly to the  $w_i$ 's, however the constraints in Eq. (17) apply to the  $a_{i,j}$ 's and cannot be easily applied to the  $\alpha_k$ 's. Nevertheless, in practice, it is frequently sufficient to weakly enforce the constraints in Eq. (17) by enforcing the following constraints on the  $\alpha_k$ 's

$$1 \geq \alpha_k \geq 0. \quad (36)$$

These constraints do not ensure that the quadrature points remain inside of the  $d$ -simplex, but they encourage this by ensuring that all points lie inside of the  $d$ -sphere of radius  $d|\mathbf{v}_1|$  centered at the centroid ( $\mathbf{x}_0$ ) of the  $d$ -simplex.

Equality constraints on the unknowns are obtained by evaluating  $\epsilon_n$  in Eq. (14). In particular, in order to obtain a quadrature rule with truncation error of order  $\delta^n$ , it is required that all error terms of order less than  $\delta^n$  vanish, and therefore, equality constraints are obtained by insisting that all error terms  $\epsilon_m$  vanish for  $m = 0, \dots, (n-1)$ . The requirement that  $\epsilon_m = 0$  can be written in matrix form as follows

$$\epsilon_m = \delta^m (\partial f_m)^T \mathbf{C}_m \mathbf{u}_m = 0, \quad (37)$$

where  $(\partial f_m)^T$  is a vector of partial derivative terms of order  $m$ ,  $\mathbf{C}_m$  is a matrix of constant coefficients (which need not be full rank), and  $\mathbf{u}_m$  is a vector containing monomial terms of the unknown  $\alpha_k$ 's and  $w_i$ 's. In order to ensure that Eq. (37) holds, one requires that

$$\mathbf{C}_m \mathbf{u}_m = 0, \quad (38)$$

or equivalently

$$\mathcal{C}_m \mathbf{u}_m = 0, \quad (39)$$

where  $\mathcal{C}_m$  is the matrix that arises from reducing  $\mathbf{C}_m$  to row echelon form and removing all rows of zeros. Evidently  $\text{rank}(\mathcal{C}_m) = \text{rank}(\mathbf{C}_m)$  if and only if  $\mathbf{C}_m$  is full rank.

Now, the matrices  $\mathcal{C}_m$  for  $m = 0, \dots, (n-1)$  can be combined together into the matrix  $\mathcal{C}$ , and similarly the vectors  $\mathbf{u}_m$  can be combined together in order to form the vector  $\mathbf{u}$ . Thus, the final nonlinear system of equality constraints for the quadrature rule of order  $n$  becomes  $\mathcal{C}\mathbf{u} = 0$ .

Evidently, the number of rows in  $\mathcal{C}$  is equivalent to the number of equality constraints that must be satisfied. In order to ensure that it is possible for the system of equations  $\mathcal{C}\mathbf{u} = 0$  to have a solution, it is necessary for the number of unknowns ( $\alpha_k$ 's and  $w_i$ 's) to equal or exceed the number of equations. Therefore, for each of the quadrature rules obtained in this work, the value of  $n$  was chosen such that the number of unknowns was greater than or equal to the number of unique equality constraints associated with the error terms of order less than  $n$ .

Having established a methodology for forming the constraints and identifying the order of a quadrature rule  $n$ , one may proceed to form an objective function and solve the resulting optimization problem.

#### 3.2. Forming the objective function and solving the optimization problem

In following the approach of [17], the objective function  $\mathcal{J}_n$  can be formed as the sum of the squares of all unique equality constraints of degree  $n$ , i.e.,

$$\mathcal{J}_n = \mathbf{u}_n^T \mathcal{C}_n^T \mathcal{C}_n \mathbf{u}_n. \quad (40)$$



**Table 1**

Order of accuracy estimates for quadrature rules on the 2-simplex with  $N_p = 1, 3, 6, 10, 15, 21, 28, 36, 45, 55$ , and 66 points. Estimates are provided for quadrature rules that are consistent and inconsistent with the 2-SCP configurations.

$N_l$	$N_p$	Order
Rules that are consistent with 2-SCP configurations		
1	1	2
2	3	3
3	6	5
4	10	6
5	15	8
6	21	9
7	28	11
8	36	13
9	45	15
10	55	16
11	66	18
Rules that are inconsistent with 2-SCP configurations		
9	45	15
10	55	16
11	66	18

In accordance with this definition, the constrained optimization problem for each quadrature rule of degree  $n$  becomes

$$\begin{aligned}
 &\text{minimize} && \mathcal{J}_n = \mathbf{u}_n^T \mathbf{C}_n^T \mathbf{C}_n \mathbf{u}_n \\
 &\text{subject to} && \mathbf{C} \mathbf{u} = 0 \\
 &&& w_i \geq 0, \quad 1 \geq \alpha_k \geq 0.
 \end{aligned} \tag{41}$$

In order to obtain the quadrature rules in this work, Eq. (41) was solved iteratively using a sequential quadratic programming (SQP) algorithm. This algorithm was chosen due to its ability to treat equality and inequality constraints of linear and nonlinear type. In accordance with the standard SQP approach (described in [18–21]), all constraints were incorporated into a Lagrangian function, and approximate Hessians of this function (computed via the Broyden–Fletcher–Goldfarb–Shanno (BFGS) method [22–25]) were used to create a sequence of quadratic programming subproblems. The subproblems were solved using an Active-Set strategy (described in [26,27]) in order to obtain search directions, and a line search algorithm (utilizing a merit function described in [28,29]) was employed in order to identify step lengths. In turn, these step lengths and search directions were used to successively update the solution.

The converged, optimal solutions were sensitive to initial conditions, indicating that the optimization problems possessed multiple local minima. In an effort to obtain global minima,  $10^6$  initial conditions were employed during the optimization of each quadrature rule with  $N_p = 1, 3, 6, 10, 15, 21$ , and 28 points on the 2-simplex,  $10^5$  initial conditions were employed for the quadrature rules with  $N_p = 36, 45, 55$ , and 66 points on the 2-simplex, and  $2.5 \times 10^4$  initial conditions were employed for the quadrature rule with  $N_p = 84$  points on the 3-simplex. These initial conditions were obtained by subjecting the  $d$ -SCP configurations to random perturbations. The optimal quadrature rules obtained through this process are discussed in the next section.

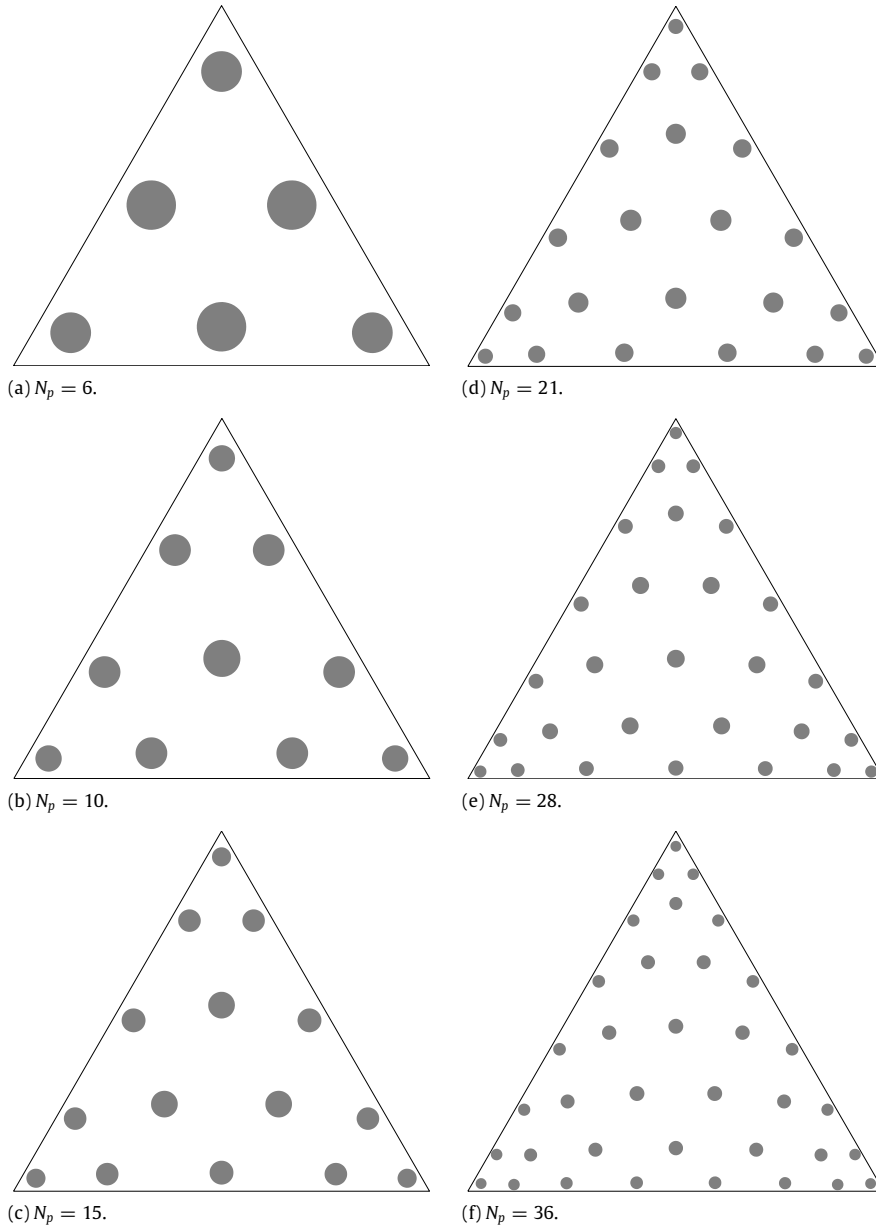
#### 4. Results of optimization

Table B.1 in Appendix B contains a set of optimal quadrature rules on the 2-simplex with  $N_p = 1, 3, 6, 10, 15, 21, 28$ , and 36 points. For the sake of brevity, additional optimal quadrature rules on the 2-simplex with  $N_p = 45, 55$ , and 66 points are accumulated in Table C.1 of [30]. Note that the quadrature rule parameters in each of the tables are given to fifteen decimal places, in accordance with the limits on the precision of the computations. Figs. 3–4(c) illustrate the arrangements of points for several of the rules. From the figures, it is apparent that the points are arranged symmetrically on the 2-simplex, the points lie within the interior of the 2-simplex, and all point arrangements can be partitioned into rows that are consistent with the corresponding 2-SCP configurations.

For the sake of completeness, the requirement on consistency with the 2-SCP configurations was relaxed, and an alternate set of quadrature rules with  $N_p = 45, 55$ , and 66 points was obtained on the 2-simplex. These quadrature rules (summarized by Table C.2 of [30] and Fig. 4(d)–(f)) are more optimal than their counterparts in Table C.1, as they produce roughly an additional order of magnitude decrease in the objective function.

Nevertheless, both sets of quadrature rules have the same theoretical order of truncation error (cf. Table 1), as the magnitude of the objective function only governs the size of the constant that multiplies the truncation error term, and not the order of the term itself. Recall that the order of the truncation error term is determined by the number of degrees of freedom and the number of nonlinear constraints, which are identical for both sets of quadrature rules.

Based on the truncation errors in Table 1, the quadrature rules in Tables B.1, C.1, and C.2 are theoretically capable of exactly integrating polynomials of up to degree 17 on the 2-simplex.

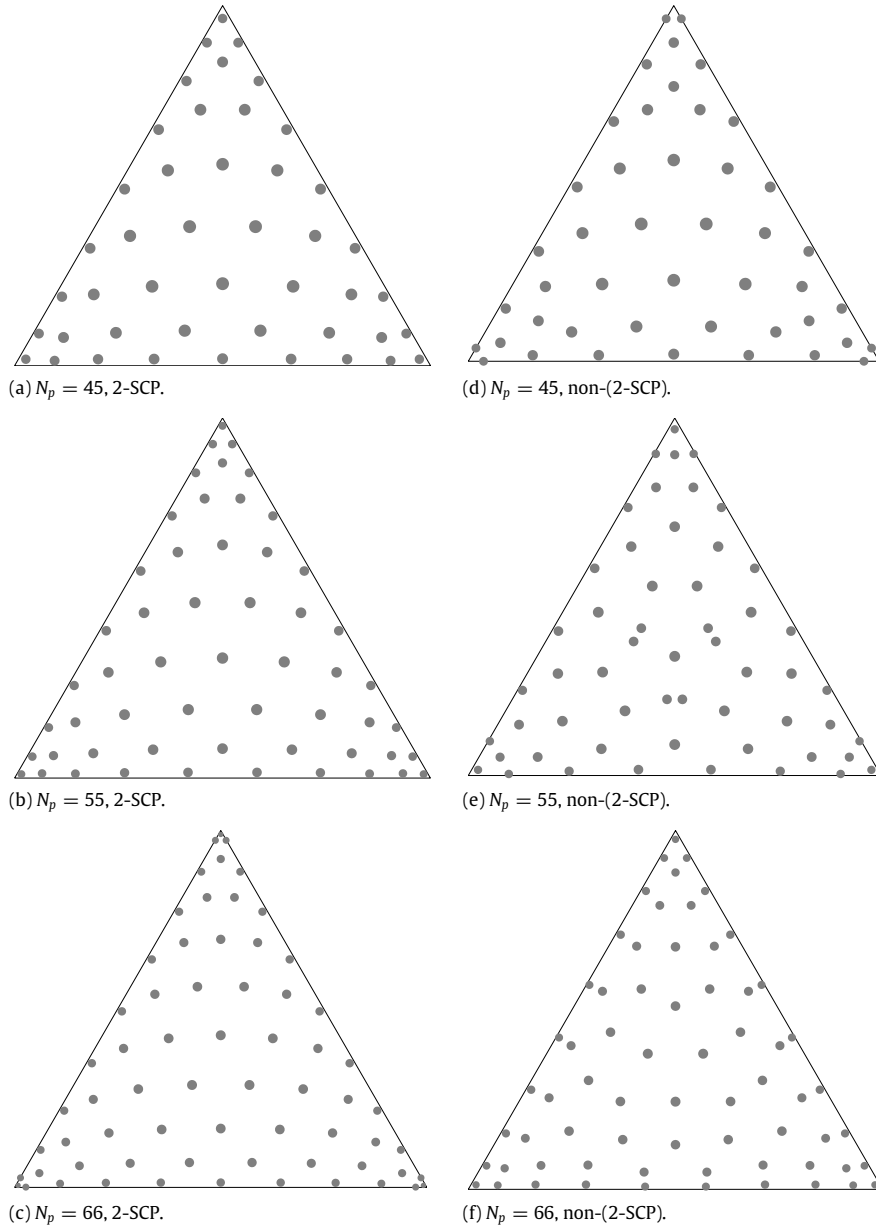


**Fig. 3.** Quadrature point locations for the quadrature rules with 2-SCP structure and  $N_p = 6, 10, 15, 21, 28$ , and  $36$  points for the triangle (the 2-simplex). The size of each point is scaled by the absolute value of the logarithm of the associated quadrature weight.

Finally, Table B.2 contains the optimal quadrature rule with  $N_p = 84$  points on the 3-simplex. Fig. 5 illustrates this rule. The truncation error for this rule was found to be  $O(\delta^{10})$ , enabling it to (in theory) exactly integrate polynomials of degree  $\leq 9$  on the 3-simplex.

## 5. Numerical experiments

A series of experiments were performed in order to assess how well the quadrature rules behaved in practice. In particular, the quadrature rules were employed to approximate integrals of various monomial terms on the unit triangular domain with vertices at  $(0, 0)$ ,  $(1, 0)$ , and  $(0, 1)$ , and the unit tetrahedral domain with vertices at  $(0, 0, 0)$ ,  $(1, 0, 0)$ ,  $(0, 1, 0)$ , and  $(0, 0, 1)$ . The unit triangular and tetrahedral domains were chosen because there are convenient expressions for the integrals of arbitrary monomial terms on these domains. Specifically, monomial terms of degree  $i + j = m$  on the unit



**Fig. 4.** Quadrature point locations for the quadrature rules with 2-SCP and non-(2-SCP) structure and  $N_p = 45, 55$ , and  $66$  points for the triangle (the 2-simplex). The size of each point is scaled by the absolute value of the logarithm of the associated quadrature weight.

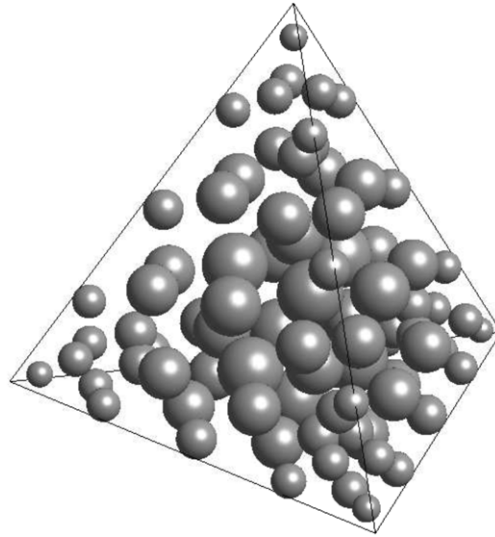
triangle have the following exact integration formula

$$\int_0^1 \int_0^{1-x} x^i y^j dy dx = \frac{i! j!}{(i+j+2)!}, \quad (42)$$

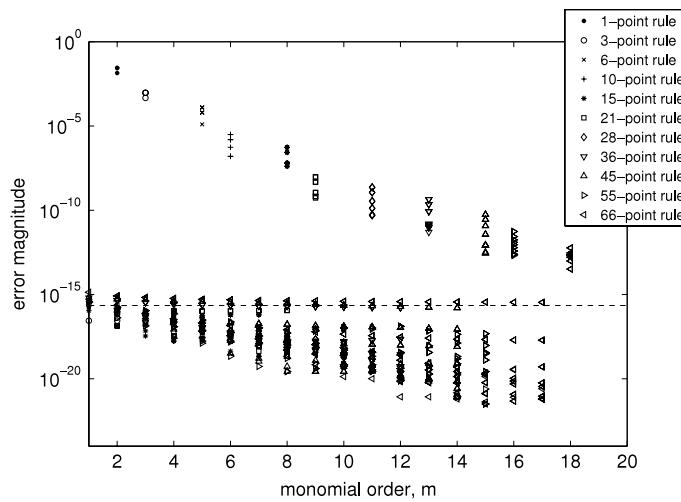
and similarly, monomial terms of degree  $i+j+k = m$  on the unit tetrahedron have the following exact integration formula

$$\int_0^1 \int_0^{1-x} \int_0^{1-x-y} x^i y^j z^k dz dy dx = \frac{i! j! k!}{(i+j+k+3)!}. \quad (43)$$

The quadrature rules from this work were employed to integrate all  $(m+1)(m+2)/2$  monomial terms of degree  $\leq m$  in 2D, and all  $(m+1)(m+2)(m+3)/6$  monomial terms of degree  $\leq m$  in 3D, where the order  $m$  was allowed to vary from 0 to 18 in 2D, and from 0 to 10 in 3D. The resulting errors associated with each quadrature rule are shown in Figs. 6–8.



**Fig. 5.** Quadrature point locations for the quadrature rule with  $N_p = 84$  points for the tetrahedron (the 3-simplex). The size of each point is scaled by the associated quadrature weight.



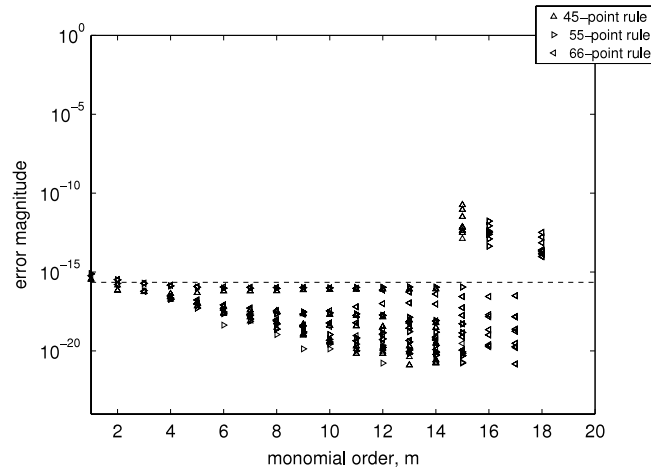
**Fig. 6.** Errors for integrating individual monomials over the unit triangle using the quadrature rules with 2-SCP structure. Machine precision is denoted by the dashed line.

From Figs. 6–8, it is clear that each quadrature rule achieves exact integration (to within machine precision) of all monomial terms whose order is strictly less than that of the truncation error. In other words, each quadrature rule is capable of exactly integrating a polynomial of the expected degree.

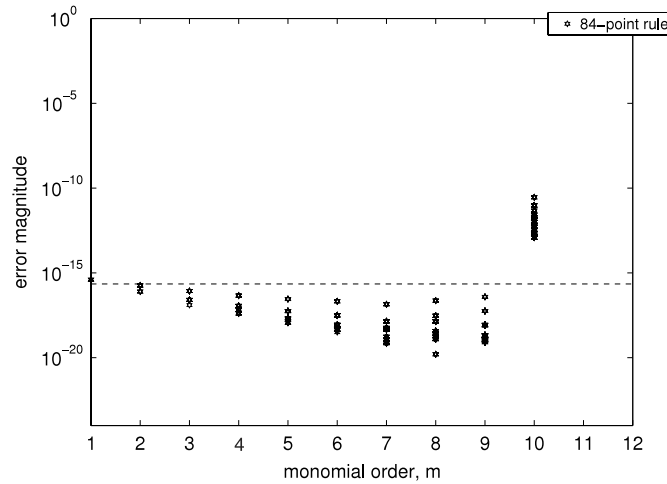
Another set of experiments was performed in order to assess the orders of accuracy of the quadrature rules. For these experiments, the quadrature rules were employed to integrate the following  $m$ th-degree polynomial functions in 2D and 3D

$$g_{2D} = \frac{\sum_{i=0}^m \sum_{j=0}^{m-i} 2(i+1)(j+1)x^i y^j}{(m+1)(m+2)}, \quad (44)$$

$$g_{3D} = \frac{\sum_{i=0}^m \sum_{j=0}^{m-i} \sum_{k=0}^{m-i-j} 6(i+1)(j+1)(k+1)x^i y^j z^k}{(m+1)(m+2)(m+3)}. \quad (45)$$



**Fig. 7.** Errors for integrating individual monomials over the unit triangle using the quadrature rules with non-(2-SCP) structure. Machine precision is denoted by the dashed line.



**Fig. 8.** Errors for integrating individual monomials over the unit tetrahedron. Machine precision is denoted by the dashed line.

Polynomial orders of  $m = 60$  and  $m = 40$  were used in 2D and 3D, respectively. In addition, the quadrature rules were utilized to integrate the following 2D and 3D transcendental functions

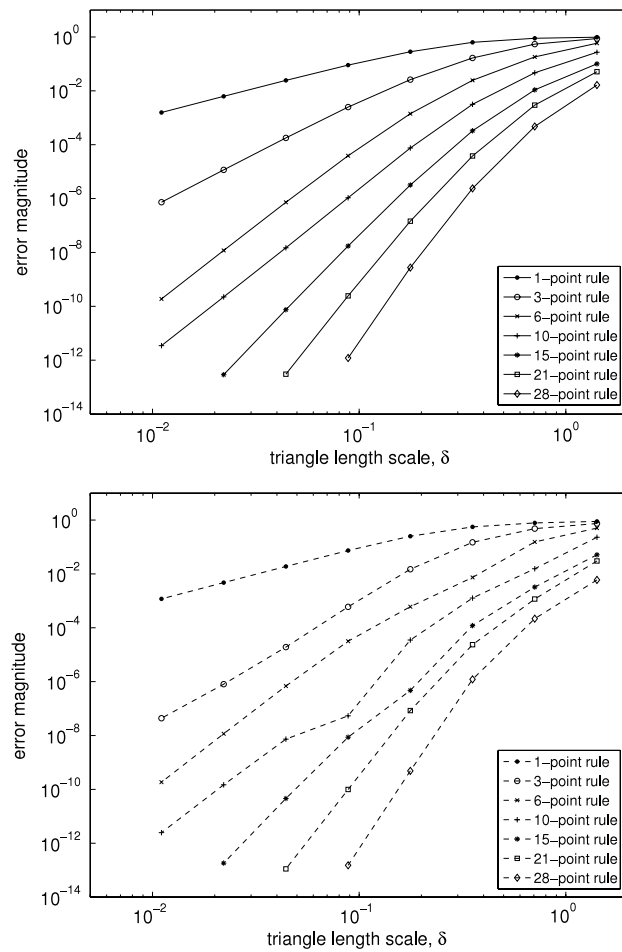
$$\tilde{g}_{2D} = \sum_{k=1}^m \left\{ \frac{(k\pi)^2 (\exp[k(x+y)] + \sin(k\pi x) \sin(k\pi y))}{m(1 + \pi^2 (\exp[k] - 1)^2 + \cos(k\pi) (\cos(k\pi) - 2))} \right\}, \quad (46)$$

$$\tilde{g}_{3D} = \sum_{k=1}^m \left\{ \frac{(k\pi)^3 (\exp[k(x+y+z)] + \sin(k\pi x) \sin(k\pi y) \sin(k\pi z))}{m(\pi^3 (\exp[k] - 1)^3 + 8 \sin^6(k\pi/2))} \right\}. \quad (47)$$

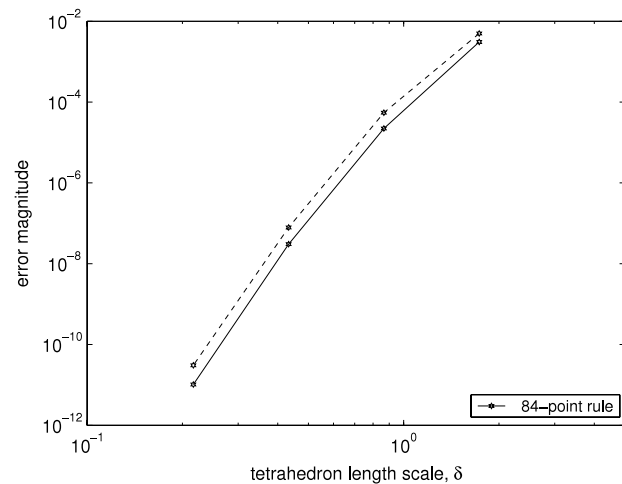
For these transcendental functions, values of  $m = 45$  and  $m = 20$  were used in 2D and 3D, respectively.

Integration of  $g_{2D}$ ,  $g_{3D}$ ,  $\tilde{g}_{2D}$ , and  $\tilde{g}_{3D}$  was performed on the unit square  $[0, 1] \times [0, 1]$  and the unit cube  $[0, 1] \times [0, 1] \times [0, 1]$ . It should be noted that the coefficients on the terms in Eqs. (44)–(47) have a normalizing effect, ensuring that the integrals of  $g_{2D}$ ,  $g_{3D}$ ,  $\tilde{g}_{2D}$ , and  $\tilde{g}_{3D}$  over the unit square and unit cube have unit values.

The experiments were performed on regular triangular grids with  $N_{tri} = 2, 8, 32, 128, 512, 2048, 8192$ , and  $32768$  elements, and on regular tetrahedral grids with  $N_{tet} = 6, 48, 384$ , and  $3072$  elements. On these grids, the order of accuracy of the truncation error was assessed for the quadrature rules with  $N_p = 1, 3, 6, 10, 15, 21$ , and  $28$  points on the 2-simplex and for the rule with  $N_p = 84$  points on the 3-simplex. Note that the results were frequently obtained on coarser grids for the higher order quadrature rules with  $N_p = 15, 21$ , and  $28$  points, as the results on the finer grids were polluted by round-off errors. Furthermore, note that results could not be obtained for the high-order quadrature rules with  $N_p = 36, 45, 55$ , and  $66$  on the 2-simplex, because the magnitudes of the errors reached the level of machine zero too rapidly during the course of grid refinement, and therefore were polluted by roundoff errors.



**Fig. 9.** Numerical orders of accuracy of quadrature rules on triangular grids for the polynomial function  $g_{2D}$  (top, solid lines) and the transcendental function  $\tilde{g}_{2D}$  (bottom, dashed lines).



**Fig. 10.** Numerical orders of accuracy of a quadrature rule on tetrahedral grids for the polynomial function  $g_{3D}$  (solid line) and the transcendental function  $\tilde{g}_{3D}$  (dashed line).

Fig. 9 and Table 2 show the orders of accuracy for the quadrature rules with  $N_p = 1, 3, 6, 10, 15, 21$ , and 28 points on the 2-simplex and Fig. 10 and Table 3 show the order of accuracy for the quadrature rule with  $N_p = 84$  points on the 3-simplex. From the figures and tables, it is clear that (at the very least) the expected order of accuracy is obtained for each quadrature rule, and that in some cases, the order of accuracy exceeds expectations.

**Table 2**

Numerical orders of accuracy for quadrature rules on the 2-simplex with  $N_p = 1, 3, 6, 10, 15, 21$ , and 28 points for the polynomial function  $g_{2D}$  and the transcendental function  $\tilde{g}_{2D}$ .

$N_l$	$N_p$	$N_{tri}$	$\delta$	Error magnitude $g$	Order $g$	Error magnitude $\tilde{g}$	Order $\tilde{g}$
1	1	2	1.414e+00	9.772e-01		8.914e-01	
		8	7.071e-01	8.964e-01	0.125	7.792e-01	0.194
		32	3.536e-01	6.311e-01	0.506	5.542e-01	0.492
		128	1.768e-01	2.821e-01	1.162	2.499e-01	1.149
		512	8.839e-02	9.010e-02	1.647	7.363e-02	1.763
		2048	4.419e-02	2.439e-02	1.885	1.885e-02	1.965
		8192	2.210e-02	6.232e-03	1.969	4.728e-03	1.996
2	3	32768	1.105e-02	1.567e-03	1.992	1.183e-03	1.999
		2	1.414e+00	8.695e-01		7.418e-01	
		8	7.071e-01	5.427e-01	0.680	4.792e-01	0.631
		32	3.536e-01	1.657e-01	1.711	1.485e-01	1.690
		128	1.768e-01	2.611e-02	2.666	1.485e-02	3.322
		512	8.839e-02	2.492e-03	3.389	5.962e-04	4.639
		2048	4.419e-02	1.790e-04	3.799	1.920e-05	4.957
3	6	8192	2.210e-02	1.162e-05	3.945	8.074e-07	4.571
		32768	1.105e-02	7.331e-07	3.986	4.378e-08	4.205
		2	1.414e+00	5.838e-01		5.012e-01	
		8	7.071e-01	1.818e-01	1.683	1.538e-01	1.704
		32	3.536e-01	2.451e-02	2.891	7.366e-03	4.384
		128	1.768e-01	1.415e-03	4.114	6.062e-04	3.603
		512	8.839e-02	3.892e-05	5.184	3.163e-05	4.260
4	10	2048	4.419e-02	7.265e-07	5.744	6.841e-07	5.531
		8192	2.210e-02	1.190e-08	5.931	1.163e-08	5.878
		32768	1.105e-02	1.883e-10	5.982	1.857e-10	5.969
		2	1.414e+00	2.713e-01		2.292e-01	
		8	7.071e-01	4.679e-02	2.536	1.562e-02	3.876
		32	3.536e-01	3.152e-03	3.892	1.266e-03	3.625
		128	1.768e-01	7.563e-05	5.381	3.513e-05	5.171
5	15	512	8.839e-02	1.071e-06	6.142	5.392e-08	9.348
		2048	4.419e-02	1.490e-08	6.167	7.349e-09	2.875
		8192	2.210e-02	2.231e-10	6.062	1.485e-10	5.629
		32768	1.105e-02	3.466e-12	6.008	2.478e-12	5.905
		2	1.414e+00	1.007e-01		5.102e-02	
		8	7.071e-01	1.085e-02	3.214	3.242e-03	3.976
		32	3.536e-01	3.298e-04	5.040	1.204e-04	4.751
6	21	128	1.768e-01	3.200e-06	6.687	4.728e-07	7.992
		512	8.839e-02	1.738e-08	7.525	8.652e-09	5.772
		2048	4.419e-02	7.497e-11	7.857	4.585e-11	7.560
		8192	2.210e-02	2.899e-13	8.015	1.840e-13	7.961
		2	1.414e+00	5.106e-02		3.011e-02	
		8	7.071e-01	2.941e-03	4.118	1.165e-03	4.692
		32	3.536e-01	3.820e-05	6.267	2.350e-05	5.632
7	28	128	1.768e-01	1.447e-07	8.044	8.415e-08	8.125
		512	8.839e-02	2.451e-10	9.206	9.915e-11	9.729
		2048	4.419e-02	3.033e-13	9.659	1.118e-13	9.792
		2	1.414e+00	1.640e-02		5.962e-03	
		8	7.071e-01	4.760e-04	5.107	2.185e-04	4.770
		32	3.536e-01	2.383e-06	7.642	1.214e-06	7.492
		128	1.768e-01	2.741e-09	9.764	4.858e-10	11.287
		512	8.839e-02	1.209e-12	11.147	1.519e-13	11.643

**Table 3**

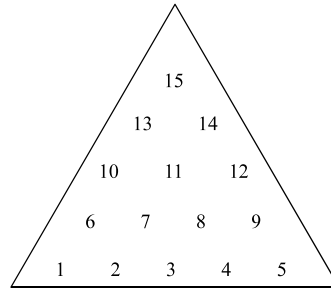
Numerical orders of accuracy for the quadrature rule on the 3-simplex with  $N_p = 84$  points for the polynomial function  $g_{3D}$  and the transcendental function  $\tilde{g}_{3D}$ .

$N_l$	$N_p$	$N_{tet}$	$\delta$	Error magnitude $g$	Order $g$	Error magnitude $\tilde{g}$	Order $\tilde{g}$
7	84	6	1.732e+00	3.074e-03		4.990e-03	
		48	8.660e-01	2.220e-05	7.113	5.453e-05	6.516
		384	4.330e-01	3.046e-08	9.509	7.890e-08	9.433
		3072	2.165e-01	1.021e-11	11.543	3.049e-11	11.337

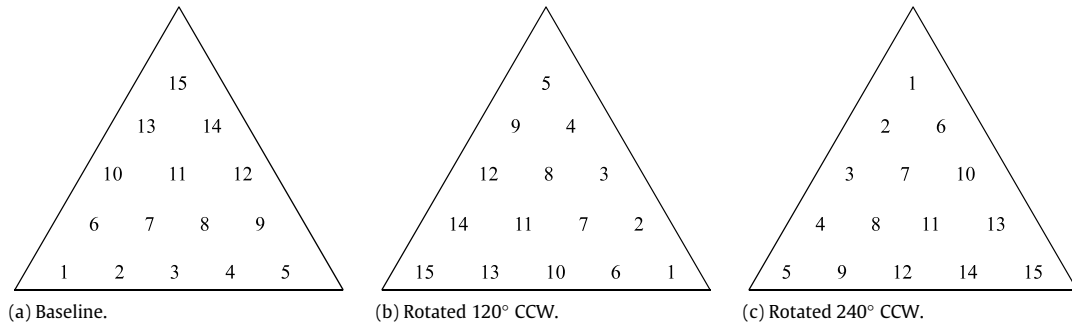
## 6. Conclusion

This work has identified a collection of optimal quadrature rules with  $N_p = 1, 3, 6, 10, 15, 21, 28, 36, 45, 55$ , and 66 points on the 2-simplex and  $N_p = 84$  points on the 3-simplex. These quadrature rules were shown to be symmetric, possess





**Fig. A.1.** Baseline convention for numbering quadrature points on a triangle for the case of  $N_p = 15$ .



**Fig. A.2.** Potential orientations of quadrature point numberings on a triangle for the case of  $N_p = 15$ .

positive quadrature weights, and have all quadrature points located within the interiors of the simplexes. Furthermore, the majority of the quadrature rules were shown to maintain a SCP-type structure, which facilitates the formulation of indexing conventions and mappings that arise in the context of finite element methods. Finally, numerical experiments verified that the quadrature rules are capable of accurately integrating high-degree polynomial functions and highly-nonlinear transcendental functions. Taken together, these favorable properties of the quadrature rules make them particularly well-suited for applications in the fields of mechanical engineering, aeronautical engineering, and computational physics.

## Appendix A. Mapping the indices of quadrature points on the 2-simplex

When discontinuous finite element methods are employed on 3D meshes of simplex elements, the solution is (in general) double-valued at the 2-simplex interfaces between the 3-simplex elements, as the solution on the faces of each element is not constrained to be identical to the solution on the faces of neighboring elements. Because of the double-valued nature of the solution, it is convenient to integrate over faces in the mesh by defining sets of quadrature points associated with each face of each element in the mesh, where it should be noted that the quadrature points on adjacent faces must be collocated. However, while these points are collocated, they are not guaranteed to be ordered (index-wise) in the same way, as their indexing may depend on the orientation of the element with which they are associated. Thus, it may be necessary to construct mappings between indices of the collocated points in order to determine the appropriate pairs of solution values at these points, and to enable the numerical integration of single-valued functions of the pairs of solution values at these points.

In practice, creating a mapping between the indices of the collocated points on a pair of adjacent 2-simplex faces is a simpler task if the points on each 2-simplex are arranged in a layered structure consistent with the corresponding 2-SCP configuration. In this case, the points can be numbered in increasing order from left to right and bottom to top. An example of the numbering convention is shown in Fig. A.1 for the case of  $N_p = 15$ .

In general, the index  $i = 0, \dots, (N_p - 1)$  for each point can be computed based on the number of layers  $N_l$  via the following pseudo code

```

for  $j = 0 \rightarrow (N_l - 1)$  do
  for  $k = 0 \rightarrow (N_l - j - 1)$  do
     $i \leftarrow jN_l - \left\lfloor \frac{j(j-1)}{2} \right\rfloor + k$ 
  end for
end for

```

**Table B.1**

Quadrature rules with  $N_p = 1, 3, 6, 10, 15, 21, 28$ , and 36 points for the triangle (the 2-simplex). Note that the quadrature point locations for these rules are consistent with the 2-SCP configurations.

$N_i$	$i$	$a_{i,1}$	$a_{i,2}$	$a_{i,3}$	$w_i$
1	1	0.333333333333333	0.333333333333333	0.333333333333333	1.000000000000000
2	1	0.666666666666667	0.166666666666667	0.166666666666667	0.333333333333333
	2	0.166666666666667	0.666666666666667	0.166666666666667	0.333333333333333
	3	0.166666666666667	0.166666666666667	0.666666666666667	0.333333333333333
3	1	0.816847572980440	0.091576213509780	0.091576213509780	0.109951743655333
	2	0.091576213509780	0.816847572980440	0.091576213509780	0.109951743655333
	3	0.091576213509780	0.091576213509780	0.816847572980440	0.109951743655333
	4	0.445948490915964	0.445948490915964	0.108103018168071	0.223381589678000
	5	0.445948490915964	0.108103018168071	0.445948490915964	0.223381589678000
	6	0.108103018168071	0.445948490915964	0.445948490915964	0.223381589678000
4	1	0.888871894660413	0.055564052669793	0.055564052669793	0.041955512996649
	2	0.055564052669793	0.888871894660413	0.055564052669793	0.041955512996649
	3	0.055564052669793	0.055564052669793	0.888871894660413	0.041955512996649
	4	0.295533711735893	0.634210747745723	0.070255540518384	0.112098412070887
	5	0.295533711735893	0.070255540518384	0.634210747745723	0.112098412070887
	6	0.070255540518384	0.295533711735893	0.634210747745723	0.112098412070887
	7	0.634210747745723	0.295533711735893	0.070255540518384	0.112098412070887
	8	0.634210747745723	0.070255540518384	0.295533711735893	0.112098412070887
	9	0.070255540518384	0.634210747745723	0.295533711735893	0.112098412070887
	10	0.333333333333333	0.333333333333333	0.333333333333333	0.201542988584730
5	1	0.928258244608533	0.035870877695734	0.035870877695734	0.017915455012303
	2	0.035870877695734	0.928258244608533	0.035870877695734	0.017915455012303
	3	0.035870877695734	0.035870877695734	0.928258244608533	0.017915455012303
	4	0.516541208464066	0.241729395767967	0.241729395767967	0.127712195881265
	5	0.241729395767967	0.516541208464066	0.241729395767967	0.127712195881265
	6	0.241729395767967	0.241729395767967	0.516541208464066	0.127712195881265
	7	0.474308787777079	0.474308787777079	0.051382424445843	0.076206062385535
	8	0.474308787777079	0.051382424445843	0.474308787777079	0.076206062385535
	9	0.051382424445843	0.474308787777079	0.474308787777079	0.076206062385535
	10	0.201503881881800	0.751183631106484	0.047312487011716	0.055749810027115
	11	0.201503881881800	0.047312487011716	0.751183631106484	0.055749810027115
	12	0.047312487011716	0.201503881881800	0.751183631106484	0.055749810027115
	13	0.751183631106484	0.201503881881800	0.047312487011716	0.055749810027115
	14	0.751183631106484	0.047312487011716	0.201503881881800	0.055749810027115
	15	0.047312487011716	0.751183631106484	0.201503881881800	0.055749810027115
6	1	0.943774095634672	0.028112952182664	0.028112952182664	0.010359374696538
	2	0.028112952182664	0.943774095634672	0.028112952182664	0.010359374696538
	3	0.028112952182664	0.028112952182664	0.943774095634672	0.010359374696538
	4	0.645721803061365	0.177139098469317	0.177139098469317	0.075394884326738
	5	0.177139098469317	0.645721803061365	0.177139098469317	0.075394884326738
	6	0.177139098469317	0.177139098469317	0.645721803061365	0.075394884326738
	7	0.405508595867433	0.405508595867433	0.188982808265134	0.097547802373242
	8	0.405508595867433	0.188982808265134	0.405508595867433	0.097547802373242
	9	0.188982808265134	0.405508595867433	0.405508595867433	0.097547802373242
	10	0.148565812270887	0.817900980028499	0.033533207700614	0.028969269372473
	11	0.148565812270887	0.033533207700614	0.817900980028499	0.028969269372473
	12	0.033533207700614	0.148565812270887	0.817900980028499	0.028969269372473
	13	0.817900980028499	0.148565812270887	0.033533207700614	0.028969269372473
	14	0.817900980028499	0.033533207700614	0.148565812270887	0.028969269372473
	15	0.033533207700614	0.817900980028499	0.148565812270887	0.028969269372473
	16	0.357196298615681	0.604978911775132	0.037824789609186	0.046046366595935
	17	0.357196298615681	0.037824789609186	0.604978911775132	0.046046366595935
	18	0.037824789609186	0.357196298615681	0.604978911775132	0.046046366595935
	19	0.604978911775132	0.357196298615681	0.037824789609186	0.046046366595935
	20	0.604978911775132	0.037824789609186	0.357196298615681	0.046046366595935
	21	0.037824789609186	0.604978911775132	0.357196298615681	0.046046366595935
7	1	0.960045625755613	0.019977187122193	0.019977187122193	0.005272170280495
	2	0.019977187122193	0.960045625755613	0.019977187122193	0.005272170280495
	3	0.019977187122193	0.019977187122193	0.960045625755613	0.005272170280495
	4	0.736556464940005	0.131721767529998	0.131721767529998	0.044552936679504
	5	0.131721767529998	0.736556464940005	0.131721767529998	0.044552936679504
	6	0.131721767529998	0.131721767529998	0.736556464940005	0.044552936679504
	7	0.333333333333333	0.333333333333333	0.333333333333333	0.083608212215637
	8	0.485135346793461	0.485135346793461	0.029729306413079	0.033815712804198
	9	0.485135346793461	0.029729306413079	0.485135346793461	0.033815712804198
	10	0.029729306413079	0.485135346793461	0.485135346793461	0.033815712804198

Table B.1 (continued)

$N_i$	$i$	$a_{i,1}$	$a_{i,2}$	$a_{i,3}$	$w_i$
8	11	0.107951981846011	0.867911210117951	0.024136808036039	0.015710461340183
	12	0.107951981846011	0.024136808036039	0.867911210117951	0.015710461340183
	13	0.024136808036039	0.107951981846011	0.867911210117951	0.015710461340183
	14	0.867911210117951	0.107951981846011	0.024136808036039	0.015710461340183
	15	0.867911210117951	0.024136808036039	0.107951981846011	0.015710461340183
	16	0.024136808036039	0.867911210117951	0.107951981846011	0.015710461340183
	17	0.270840772921567	0.700872570380723	0.028286656697710	0.028205136280616
	18	0.270840772921567	0.028286656697710	0.700872570380723	0.028205136280616
	19	0.028286656697710	0.270840772921567	0.700872570380723	0.028205136280616
	20	0.700872570380723	0.270840772921567	0.028286656697710	0.028205136280616
	21	0.700872570380723	0.028286656697710	0.270840772921567	0.028205136280616
	22	0.028286656697710	0.700872570380723	0.270840772921567	0.028205136280616
	23	0.316549598844617	0.536654684206138	0.146795716949245	0.066995957127830
	24	0.316549598844617	0.146795716949245	0.536654684206138	0.066995957127830
	25	0.146795716949245	0.316549598844617	0.536654684206138	0.066995957127830
	26	0.536654684206138	0.316549598844617	0.146795716949245	0.066995957127830
	27	0.536654684206138	0.146795716949245	0.316549598844617	0.066995957127830
	28	0.146795716949245	0.536654684206138	0.316549598844617	0.066995957127830
	1	0.957657154441070	0.021171422779465	0.021171422779465	0.005639123786910
	2	0.021171422779465	0.957657154441070	0.021171422779465	0.005639123786910
	3	0.021171422779465	0.021171422779465	0.957657154441070	0.005639123786910
	4	0.798831205208225	0.100584397395888	0.100584397395888	0.027148968192278
	5	0.100584397395888	0.798831205208225	0.100584397395888	0.027148968192278
	6	0.100584397395888	0.100584397395888	0.798831205208225	0.027148968192278
	7	0.457923384576135	0.271038307711932	0.271038307711932	0.063100912533359
	8	0.271038307711932	0.457923384576135	0.271038307711932	0.063100912533359
	9	0.271038307711932	0.271038307711932	0.457923384576135	0.063100912533359
	10	0.440191258403832	0.440191258403832	0.119617483192335	0.051752795679899
	11	0.440191258403832	0.119617483192335	0.440191258403832	0.051752795679899
	12	0.119617483192335	0.440191258403832	0.440191258403832	0.051752795679899
	13	0.101763679498021	0.879979641427232	0.018256679074748	0.009866753574646
	14	0.101763679498021	0.018256679074748	0.879979641427232	0.009866753574646
	15	0.018256679074748	0.101763679498021	0.879979641427232	0.009866753574646
	16	0.879979641427232	0.101763679498021	0.018256679074748	0.009866753574646
	17	0.879979641427232	0.018256679074748	0.101763679498021	0.009866753574646
	18	0.018256679074748	0.879979641427232	0.101763679498021	0.009866753574646
	19	0.394033271669987	0.582562022863673	0.023404705466341	0.022008204800147
	20	0.394033271669987	0.023404705466341	0.582562022863673	0.022008204800147
	21	0.023404705466341	0.394033271669987	0.582562022863673	0.022008204800147
	22	0.582562022863673	0.394033271669987	0.023404705466341	0.022008204800147
	23	0.582562022863673	0.023404705466341	0.394033271669987	0.022008204800147
	24	0.023404705466341	0.582562022863673	0.394033271669987	0.022008204800147
	25	0.226245530909229	0.751530614542782	0.022223854547989	0.016644570076736
	26	0.226245530909229	0.022223854547989	0.751530614542782	0.016644570076736
	27	0.022223854547989	0.226245530909229	0.751530614542782	0.016644570076736
	28	0.751530614542782	0.226245530909229	0.022223854547989	0.016644570076736
	29	0.751530614542782	0.022223854547989	0.226245530909229	0.016644570076736
	30	0.022223854547989	0.751530614542782	0.226245530909229	0.016644570076736
	31	0.635737183263105	0.249079227621332	0.115183589115563	0.044326238118914
	32	0.635737183263105	0.115183589115563	0.249079227621332	0.044326238118914
	33	0.115183589115563	0.635737183263105	0.249079227621332	0.044326238118914
	34	0.249079227621332	0.635737183263105	0.115183589115563	0.044326238118914
	35	0.249079227621332	0.115183589115563	0.635737183263105	0.044326238118914
	36	0.115183589115563	0.249079227621332	0.635737183263105	0.044326238118914

Now, the points (with index  $i$ ) must be related to the collocated points (with index  $i_n$ ) on the adjacent face. In order to relate these points, one must consider all three possible orientations of points (with indices  $i_0$ ,  $i_1$ , and  $i_2$ ) on the adjacent face. Fig. A.2, shows an example of the three possible orientations of points on the adjacent face for the case of  $N_p = 15$ .

In general, the points with indices  $i_0$ ,  $i_1$ , or  $i_2$  can be mapped to the appropriate (collocated) points with index  $i$  via the mappings  $M_{i_0}$ ,  $M_{i_1}$ , and  $M_{i_2}$  that are defined (in pseudocode) as follows

```

for  $j = 0 \rightarrow (N_i - 1)$  do
  for  $k = 0 \rightarrow (N_i - j - 1)$  do
     $M_{i_2} \leftarrow jN_i - \left\lfloor \frac{j(j-1)}{2} \right\rfloor + N_i - j - k - 1$ 
  end for
end for

```

**Table B.2**Quadrature rule with  $N_p = 84$  points for the tetrahedron (the 3-simplex).

$N_i$	$i$	$a_{i,1}$	$a_{i,2}$	$a_{i,3}$	$a_{i,4}$	$w_i$
7	1	0.919364576755549	0.026878474414817	0.026878474414817	0.026878474414817	0.002144935144316
	2	0.026878474414817	0.919364576755549	0.026878474414817	0.026878474414817	0.002144935144316
	3	0.026878474414817	0.026878474414817	0.919364576755549	0.026878474414817	0.002144935144316
	4	0.026878474414817	0.026878474414817	0.026878474414817	0.919364576755549	0.002144935144316
	5	0.438577972589591	0.187140675803470	0.187140675803470	0.187140675803470	0.020826641690769
	6	0.187140675803470	0.438577972589591	0.187140675803470	0.187140675803470	0.020826641690769
	7	0.187140675803470	0.187140675803470	0.438577972589591	0.187140675803470	0.020826641690769
	8	0.187140675803470	0.187140675803470	0.187140675803470	0.438577972589591	0.020826641690769
	9	0.473575835127937	0.473575835127937	0.026424164872063	0.026424164872063	0.007210136064455
	10	0.473575835127937	0.026424164872063	0.473575835127937	0.026424164872063	0.007210136064455
	11	0.473575835127937	0.026424164872063	0.026424164872063	0.473575835127937	0.007210136064455
	12	0.026424164872063	0.473575835127937	0.473575835127937	0.026424164872063	0.007210136064455
	13	0.026424164872063	0.473575835127937	0.026424164872063	0.473575835127937	0.007210136064455
	14	0.026424164872063	0.026424164872063	0.473575835127937	0.473575835127937	0.007210136064455
	15	0.352045262027356	0.352045262027356	0.147954737972644	0.147954737972644	0.030798919159712
	16	0.352045262027356	0.147954737972644	0.352045262027356	0.147954737972644	0.030798919159712
	17	0.352045262027356	0.147954737972644	0.147954737972644	0.352045262027356	0.030798919159712
	18	0.147954737972644	0.352045262027356	0.352045262027356	0.147954737972644	0.030798919159712
	19	0.147954737972644	0.352045262027356	0.147954737972644	0.352045262027356	0.030798919159712
	20	0.147954737972644	0.147954737972644	0.352045262027356	0.352045262027356	0.030798919159712
	21	0.225783205866940	0.732309909692947	0.020953442220056	0.020953442220056	0.004357844813864
	22	0.225783205866940	0.020953442220056	0.732309909692947	0.020953442220056	0.004357844813864
	23	0.225783205866940	0.020953442220056	0.020953442220056	0.732309909692947	0.004357844813864
	24	0.020953442220056	0.732309909692947	0.732309909692947	0.020953442220056	0.004357844813864
	25	0.020953442220056	0.225783205866940	0.020953442220056	0.732309909692947	0.004357844813864
	26	0.020953442220056	0.020953442220056	0.225783205866940	0.732309909692947	0.004357844813864
	27	0.732309909692947	0.225783205866940	0.020953442220056	0.020953442220056	0.004357844813864
	28	0.732309909692947	0.020953442220056	0.225783205866940	0.020953442220056	0.004357844813864
	29	0.732309909692947	0.020953442220056	0.020953442220056	0.225783205866940	0.004357844813864
	30	0.020953442220056	0.732309909692947	0.225783205866940	0.020953442220056	0.004357844813864
	31	0.020953442220056	0.732309909692947	0.020953442220056	0.225783205866940	0.004357844813864
	32	0.020953442220056	0.020953442220056	0.732309909692947	0.225783205866940	0.004357844813864
	33	0.158462939666092	0.647557594086975	0.096989733123466	0.096989733123466	0.008593530677833
	34	0.158462939666092	0.096989733123466	0.647557594086975	0.096989733123466	0.008593530677833
	35	0.158462939666092	0.096989733123466	0.096989733123466	0.647557594086975	0.008593530677833
	36	0.096989733123466	0.158462939666092	0.647557594086975	0.096989733123466	0.008593530677833
	37	0.096989733123466	0.158462939666092	0.096989733123466	0.647557594086975	0.008593530677833
	38	0.096989733123466	0.096989733123466	0.158462939666092	0.647557594086975	0.008593530677833
	39	0.647557594086975	0.158462939666092	0.096989733123466	0.096989733123466	0.008593530677833
	40	0.647557594086975	0.096989733123466	0.158462939666092	0.096989733123466	0.008593530677833
	41	0.647557594086975	0.096989733123466	0.096989733123466	0.158462939666092	0.008593530677833
	42	0.096989733123466	0.647557594086975	0.158462939666092	0.096989733123466	0.008593530677833
	43	0.096989733123466	0.647557594086975	0.096989733123466	0.158462939666092	0.008593530677833
	44	0.096989733123466	0.096989733123466	0.647557594086975	0.158462939666092	0.008593530677833
	45	0.322111431830857	0.322111431830857	0.322111431830857	0.033665704507429	0.023000681669286
	46	0.322111431830857	0.322111431830857	0.033665704507429	0.322111431830857	0.023000681669286
	47	0.322111431830857	0.033665704507429	0.322111431830857	0.322111431830857	0.023000681669286
	48	0.033665704507429	0.322111431830857	0.322111431830857	0.322111431830857	0.023000681669286
	49	0.792939256469618	0.097608162890442	0.097608162890442	0.011844417749498	0.004863063904912
	50	0.097608162890442	0.792939256469618	0.097608162890442	0.011844417749498	0.004863063904912
	51	0.097608162890442	0.097608162890442	0.792939256469618	0.011844417749498	0.004863063904912
	52	0.792939256469618	0.097608162890442	0.011844417749498	0.097608162890442	0.004863063904912
	53	0.097608162890442	0.792939256469618	0.011844417749498	0.097608162890442	0.004863063904912
	54	0.097608162890442	0.097608162890442	0.011844417749498	0.792939256469618	0.004863063904912
	55	0.792939256469618	0.011844417749498	0.097608162890442	0.097608162890442	0.004863063904912
	56	0.097608162890442	0.011844417749498	0.792939256469618	0.097608162890442	0.004863063904912
	57	0.097608162890442	0.011844417749498	0.097608162890442	0.792939256469618	0.004863063904912
	58	0.011844417749498	0.792939256469618	0.097608162890442	0.097608162890442	0.004863063904912
	59	0.011844417749498	0.097608162890442	0.792939256469618	0.097608162890442	0.004863063904912
	60	0.011844417749498	0.097608162890442	0.097608162890442	0.792939256469618	0.004863063904912
	61	0.541184412800237	0.133558160703568	0.296501020543124	0.028756405953071	0.015595140078259
	62	0.296501020543124	0.541184412800237	0.133558160703568	0.028756405953071	0.015595140078259
	63	0.133558160703568	0.296501020543124	0.541184412800237	0.028756405953071	0.015595140078259
	64	0.541184412800237	0.296501020543124	0.133558160703568	0.028756405953071	0.015595140078259
	65	0.133558160703568	0.541184412800237	0.296501020543124	0.028756405953071	0.015595140078259
	66	0.296501020543124	0.133558160703568	0.541184412800237	0.028756405953071	0.015595140078259
	67	0.541184412800237	0.133558160703568	0.028756405953071	0.296501020543124	0.015595140078259
	68	0.296501020543124	0.541184412800237	0.028756405953071	0.133558160703568	0.015595140078259
	69	0.133558160703568	0.296501020543124	0.028756405953071	0.541184412800237	0.015595140078259

Table B.2 (continued)

$N_l$	$i$	$a_{i,1}$	$a_{i,2}$	$a_{i,3}$	$a_{i,4}$	$w_i$
	70	0.541184412800237	0.296501020543124	0.028756405953071	0.133558160703568	0.015595140078259
	71	0.133558160703568	0.541184412800237	0.028756405953071	0.296501020543124	0.015595140078259
	72	0.296501020543124	0.133558160703568	0.028756405953071	0.541184412800237	0.015595140078259
	73	0.541184412800237	0.028756405953071	0.133558160703568	0.296501020543124	0.015595140078259
	74	0.296501020543124	0.028756405953071	0.541184412800237	0.133558160703568	0.015595140078259
	75	0.133558160703568	0.028756405953071	0.296501020543124	0.541184412800237	0.015595140078259
	76	0.541184412800237	0.028756405953071	0.296501020543124	0.133558160703568	0.015595140078259
	77	0.133558160703568	0.028756405953071	0.541184412800237	0.296501020543124	0.015595140078259
	78	0.296501020543124	0.028756405953071	0.133558160703568	0.541184412800237	0.015595140078259
	79	0.028756405953071	0.541184412800237	0.133558160703568	0.296501020543124	0.015595140078259
	80	0.028756405953071	0.296501020543124	0.541184412800237	0.133558160703568	0.015595140078259
	81	0.028756405953071	0.133558160703568	0.296501020543124	0.541184412800237	0.015595140078259
	82	0.028756405953071	0.541184412800237	0.296501020543124	0.133558160703568	0.015595140078259
	83	0.028756405953071	0.133558160703568	0.541184412800237	0.296501020543124	0.015595140078259
	84	0.028756405953071	0.296501020543124	0.133558160703568	0.541184412800237	0.015595140078259

```

for  $j = 0 \rightarrow (N_l - 1)$  do
  for  $k = 0 \rightarrow (N_l - j - 1)$  do
     $M_{i_1} \leftarrow \frac{N_l(N_l+1)}{2} - \left\lfloor \frac{(k+j)(k+j+1)}{2} \right\rfloor - j - 1$ 
  end for
end for

```

```

for  $j = 0 \rightarrow (N_l - 1)$  do
  for  $k = 0 \rightarrow (N_l - j - 1)$  do
     $M_{i_0} \leftarrow kN_l - \left\lfloor \frac{k(k-1)}{2} \right\rfloor + j$ 
  end for
end for

```

These mappings are convenient, as they hold for all quadrature rules for which the points are arranged in layers that are consistent with the corresponding 2-SCP configurations. For quadrature rules that do not possess this structure, the mappings do not hold, and bespoke mappings must be created. The creation of bespoke mappings is inconvenient and maybe expensive or intractable (from a practical standpoint) for quadrature rules with a large number of points. For this reason, quadrature rules with points that are arranged in layers (similar to the 2-SCP layers) are preferred.

As a final note, one should consider the following caveat. Some caution must be exercised if the quadrature points on the faces of the elements are used for interpolation purposes as well as integration purposes, as they must not only have mappings (for interface-matching purposes) but they must also be ‘well-conditioned’ for interpolation (i.e. they must have small Lebesgue constants). In [31], the favorable interpolatory conditioning of the first six quadrature rules on the 2-simplex from this article (with  $N_p = 1$  to  $N_p = 21$  points) and the six quadrature rules on the 3-simplex (due to Shunn and Ham [17]) was demonstrated. However, for several higher order rules in this article, favorable interpolation conditioning has yet to be verified, and some care should be taken if they are to be used as interpolation points. This topic will not be discussed further here, but for additional details please see [31–33].

## Appendix B. Quadrature rules on the 2-simplex and 3-simplex

See Tables B.1 and B.2.

## References

- [1] A.H. Stroud, Approximate Calculation of Multiple Integrals, Prentice-Hall, 1971.
- [2] R. Cools, P. Rabinowitz, Monomial cubature rules since Stroud: a compilation, J. Comput. Appl. Math. 48 (1993) 309–326.
- [3] R. Cools, Constructing cubature formulae: the science behind the art, Acta Numer. 6 (1997) 1–54.
- [4] R. Cools, An encyclopaedia of cubature formulas, J. Complexity 19 (2003) 445–453.
- [5] P.J. Davis, P. Rabinowitz, Methods of Numerical Integration, Courier Dover Publications, 2007.
- [6] A.H. Stroud, D. Secrest, Gaussian Quadrature Formulas, Prentice-Hall, 1966.
- [7] M.J. Graham, P. Weinacht, J. Brandeis, B.M. Irons, Engineering Applications of Numerical Integration in Stiffness Methods, Amer. Inst. Aeronaut. Astronaut. 4 (1966) 2035–2037.
- [8] M.A. Taylor, B.A. Wingate, L.P. Bos, A cardinal function algorithm for computing multivariate quadrature points, SIAM J. Numer. Anal. 45 (2007) 193–205.
- [9] M.A. Taylor, Asymmetric cubature formulas for polynomial integration in the triangle and square, J. Comput. Appl. Math. 218 (2008) 184–191.
- [10] L. Zhang, T. Cui, H. Liu, A set of symmetric quadrature rules on triangles and tetrahedra, J. Comput. Math. 27 (2009) 89–96.
- [11] H. Xiao, Z. Gimbutas, A numerical algorithm for the construction of efficient quadrature rules in two and higher dimensions, Comput. Math. Appl. 59 (2010) 663–676.

- [12] H.T. Rathod, B. Venkatesudu, K.V. Nagaraja, On the application of two symmetric Gauss Legendre quadrature rules for composite numerical integration over a tetrahedral region, *Int. J. Comput. Eng. Sci. Mech.* 7 (2006) 445–459.
- [13] H.T. Rathod, B. Venkatesudu, K.V. Nagaraja, Symmetric Gauss Legendre quadrature rules for composite numerical integration over a tetrahedral region, *J. Bull. Math.* 24 (2006) 51–79.
- [14] H.T. Rathod, K.V. Nagaraja, B. Venkatesudu, Symmetric Gauss Legendre quadrature formulas for composite numerical integration over a triangular surface, *Appl. Math. Comput.* 188 (2007) 865–876.
- [15] K.V. Nagaraja, H.T. Rathod, Symmetric Gauss Legendre quadrature rules for numerical integration over an arbitrary linear tetrahedra in Euclidean three-dimensional space, *Int. J. Math. Anal.* 4 (2010) 921–928.
- [16] J. Sarada, K.V. Nagaraja, Generalized Gaussian quadrature rules over two-dimensional regions with linear sides, *Appl. Math. Comput.* 217 (2011) 5612–5621.
- [17] L. Shunn, F. Ham, Symmetric quadrature rules for tetrahedra based on a cubic close-packed lattice arrangement, *J. Comput. Appl. Math.* 236 (2012) 4348–4364.
- [18] M. Powell, Variable metric methods for constrained optimization, *Comput. Methods Appl. Sci. Eng.* 1 (1979) 62–72.
- [19] P.E. Gill, W. Murray, M.H. Wright, *Practical Optimization*, Academic press, 1981.
- [20] R. Fletcher, *Practical Methods of Optimization*, John Wiley & Sons, 1987.
- [21] J. Nocedal, S.J. Wright, *Numerical Optimization*, Springer Verlag, 2006.
- [22] C.G. Broyden, A new double-rank minimization algorithm, *AMS Not.* 16 (1969) 670.
- [23] R. Fletcher, A new approach to variable metric algorithms, *Comput. J.* 13 (1970) 317–322.
- [24] D. Goldfarb, A family of variable metric methods derived by variational means, *Math. Comp.* 24 (1970) 23–26.
- [25] D.F. Shanno, Conditioning of quasi-Newton methods for function minimization, *Math. Comp.* 24 (1970) 647–656.
- [26] P.E. Gill, W. Murray, M.A. Saunders, M.H. Wright, Procedures for optimization problems with a mixture of bounds and general linear constraints, *ACM TOMS* 10 (1984) 282–298.
- [27] P.E. Gill, W. Murray, M.H. Wright, *Numerical Linear Algebra and Optimization*, vol. 1, Perseus Books, 1991.
- [28] S. Han, A globally convergent method for nonlinear programming, *J. Optim. Theory Appl.* 22 (1977) 297–309.
- [29] M. Powell, A fast algorithm for nonlinearly constrained optimization calculations, *Numer. Anal.* (1978) 144–157.
- [30] D.M. Williams, Energy stable high-order methods for simulating unsteady, viscous, compressible flows on unstructured grids, Ph.D. Thesis, Stanford University, 2013.
- [31] D.M. Williams, A. Jameson, Nodal Points and the Nonlinear Stability of High-Order Methods for Unsteady Flow Problems on Tetrahedral Meshes, AIAA Paper 2013-2830, 43rd AIAA Fluid Dynamics Conference, June 24–27, 2013, San Diego, CA.
- [32] B. Vioreanu, V. Rokhlin, Spectra of multiplication operators as a numerical tool, *Yale CS Tech Report* 1443, 2011.
- [33] T. Warburton, An explicit construction of interpolation nodes on the simplex, *J. Eng. Math.* 56 (2006) 247–262.