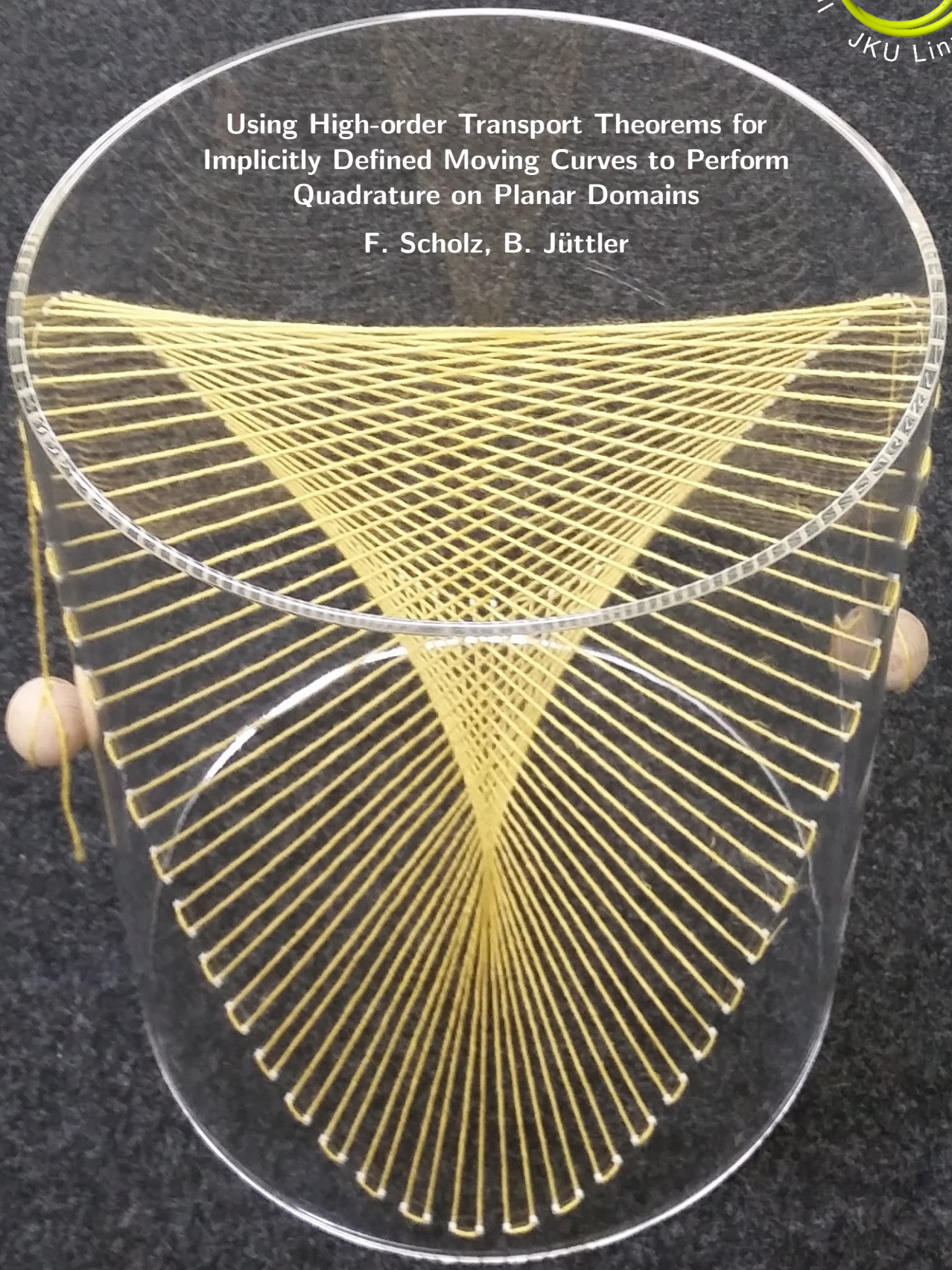
JKU

Institute of Applied Geometry
JKU Linz

Using High-order Transport Theorems for Implicitly Defined Moving Curves to Perform Quadrature on Planar Domains

F. Scholz, B. Jüttler



AG Report No. 93
April 30, 2021
www.ag.jku.at

USING HIGH-ORDER TRANSPORT THEOREMS FOR IMPLICITLY DEFINED MOVING CURVES TO PERFORM QUADRATURE ON PLANAR DOMAINS

FELIX SCHOLZ* AND BERT JÜTTLER†

Abstract. The numerical integration over a planar domain that is cut by an implicitly defined boundary curve is an important problem that arises, for example, in unfitted finite element methods and in isogeometric analysis on trimmed computational domains. In this paper, we introduce a a very general version of the transport theorem for moving domains defined by implicitly defined curves and use it to establish an efficient and accurate quadrature rule for this class of domains. In numerical experiments it is shown that the method achieves high orders of convergence. Our approach is suited for high-order geometrically unfitted finite element methods as well as for high-order trimmed isogeometric analysis.

Key words. numerical quadrature, unfitted finite element method, immersed methods, fictitious domain methods, isogeometric analysis, trimming, transport theorem

AMS subject classifications. 53A04, 65N30, 65N85

1. Introduction. Accurate and efficient quadrature rules for domains that are defined by a fixed geometry that is intersected by a curve or surface are needed in several applications.

These include geometrically unfitted finite element methods [2], such as CutFEM [6], and immersed finite element [50] and fictitious domain methods [18]. In these methods, the solution to a partial differential equation (PDE) is approximated in a discrete space that is defined by background mesh that covers the computational domain. The computational domain is then specified by its boundary, which is often represented implicitly, for example by a distance function. In general, the boundary of the background mesh does not coincide with (i.e., it is not *fitted* to) the boundary of the computational domain. Therefore, this approach results in a discretization that consists of interior elements of the background mesh as well as cut elements along the boundary of the computational domain. Dirichlet boundary conditions are imposed weakly and the stability of the discretization is ensured by modifying the bilinear form. Several approaches exist, including high-order schemes [7, 28, 34]. In order to implement unfitted FEM, a quadrature rule for the cut elements along the computational domain's boundary is needed to assemble the system matrix. According to Strang's lemma, the approximation order of the employed quadrature rule needs to match the approximation order of the discrete space. It is therefore necessary to define high-order quadrature rules for the cut elements in order to be able to employ discretization spaces of high approximation orders.

Another important application of quadrature rules on geometries that are intersected by curves appears in the framework of Isogeometric Analysis (IgA) [21]. IgA aims at approximating solutions to PDEs that are defined on models produced by Computer Aided Design (CAD) systems without changing the underlying geometric description. In CAD systems, the geometries are usually represented by tensor-product B-Splines or NURBS (non-uniform rational B-Splines). According to the isogeometric paradigm, the solution to the PDE is approximated in the same space that is use for representing the geometry. This approach enables the seamless integration of CAD and numerical analysis as well as the use of high-order discretizations, which compare favorably to finite element methods [3, 16].

In practice, CAD models increase the flexibility of tensor-product spline constructions by including *trimmed patches*. These are created when a tensor-product patch is intersected by a trimming curve or surface that splits the patch into an active and an inactive region. The advantage of this technique is that it makes it possible to represent much more complex geometries. However, in the context of numerical analysis using isogeometric analysis, the occurrence of trimmed patches results in a number of challenges [32]. Especially when using high-order spline discretizations, we need accurate quadrature rules on trimmed domains that match the approximation order of the

*Radon Institute for Computational and Applied Mathematics, Austrian Academy of Sciences, Linz/Austria (felix.scholz@ricam.oeaw.ac.at).

†Institute of Applied Geometry, Johannes Kepler University, Linz/Austria (bert.juettler@jku.at).

employed discrete spaces.

The treatment of trimmed domains has thus been a major focus of the recent research in IgA, and many similarities to unfitted FEM have been noted. Besides quadrature, several problems have been investigated: The *stability* of the discretization needs to be dealt with, since the support of the basis functions can become arbitrarily small when refining the discretization space. This issue can be resolved by employing modified bases such as extended B-Splines [22, 31], by modifying the bilinear form [4, 5], employing specialized preconditioners [12], or by the use of multi-grid solvers [13, 29]. *Adaptivity* has been introduced to isogeometric and immersed methods on trimmed domains through the use of THB-splines [10, 17]. The *coupling* between trimmed patches is a further challenge that has been addressed [40, 51], for example in the context of shells [27].

The present paper focuses entirely on quadrature. The standard approach to numerical quadrature on elements that are cut by a curve is to subdivide each element of the discretization into simple subdomains, such as polygons with curved edges, for which quadrature rules are available. It has been used extensively both for isogeometric analysis [17, 25, 48] and for unfitted FEM [14, 38]. In isogeometric analysis, the conversion of trimmed elements into a number of untrimmed entities is also referred to as *untrimming* [1, 23, 33].

Clearly, the success of this conversion depends on the robustness and the efficiency of the available tools for computing intersections and for the approximation of the cut domain's boundaries. In particular, intersection computations for free-form geometries are well known to be a hard problem [39], and the solution that has been implemented in the existing CAD technology requires a substantial amount of engineering knowledge, and sometimes even user interaction. On the one hand, the level of difficulty increases both with the order of the discretization, due to the need to generate more accurate representation of the subdomains as it is increased. On the other hand, it also increases with the degree of the geometry representations, which makes robust intersection computations more difficult. For instance, the intersection curves of bicubic patches – a standard representation for free–form geometry in CAD – possesses algebraic order 324 in the parameter domains!

Besides untrimming, a number of further approaches to the numerical integration on cut elements have been proposed.

A classical approach, which has been applied to immersed boundary methods [42], is adaptive quadrature by quadtree or octree subdivision. In this method the computational domain is subdivided adaptively along the interface until a predefined depth is reached. The method has been improved by moving the resulting quadrature nodes onto the geometric boundary in [26], resulting in the *smart octree* method. This provides accurate approximations of the integrals but the number of quadrature nodes can be very high. In [15] and [49], this method is extended by the use of error-estimators, aimed at reducing the number of sub-cells in the octree subdivision.

Another research direction aims at transforming the integrals over bivariate or trivariate domains into a number of univariate integrals. In [11] a method for the integration over tetrahedra that are cut by implicit surfaces has been presented. Each tetrahedron is parameterized with respect to suitable coordinate directions that allow for a representation of the original integral by nested univariate integrals, taking into account the appearing singularities. The reduction to univariate integrals can also be achieved via integral theorems such as the divergence theorem or Green's theorem, an idea that has been investigated extensively in different contexts [20, 41, 45]. Similar to untrimming, these methods rely on the accurate computation of intersections.

Another interesting approach builds on the construction of specific quadrature rules for each trimmed element by solving a non-linear optimization problem [36]. In order to formulate the moment-fitting equations that define the quadrature nodes, the integrals of polynomials are approximated on a subdivision of the considered element into convex sub-cells. Since solving a nonlinear optimization problem in each cell can be expensive, it has been proposed to choose the nodes in advance and only solve for the quadrature weights [24, 35].

Motivated by the marching cube algorithm, the *Corrected Linearized Trimmed quadrature rule* (CLT) for the numerical integration on planar domains intersected with implicitly defined curves was developed in [44]. It was then extended to the thee-dimensional case of volumes intersected with implicit surfaces in [43]. The method defines a fourth-order quadrature rule on

each integration cell and consists of three steps:

- In a first step it is ensured that the considered integration cell belongs to a small number of predefined base cases, which are defined by the signs of the implicit function at the cell vertices. These base cases are similar to the ones of the marching cube algorithm [37].
- In a second step the function that defines the implicit curve is approximated by a linear function, which results in the approximation of the trimming curve by a straight line. This is used as an initial approximation and by itself results in a third-order quadrature rule.
- Finally, in the third step of the method, the approximation order is increased by one by adding an error correction term that is based on the Taylor expansion of the linear interpolation between the original integral and the approximate one. This correction term consists of the first derivative of the integral over a moving domain and it is computed using the Reynolds transport theorem.

Unlike most of the existing methods, this approach does not require any intersection computations. Instead, it suffices to evaluate the function that defines the interface at the cells' vertices.

Using shape sensitivity analysis (see [9]) instead of the Reynolds transport theorem, a similar error correction approach was explored in [46]. More precisely, a *piecewise* linear (and not just linear as in our approach) approximation of the interface combined with first-order correction terms is used to generate moment-fitting equations that define quadrature rules with predefined nodes inside the computational domain. The presented experiments indicate convergence rate 2 for dimension $d = 2$, but (surprisingly) rate 4 for $d = 3$, according to Table 2 of [46] and Table 2 of [47], while our results would justify rate 3 in both cases. We believe that the results of the present paper prepare the ground for extending the moment-fitting approach to higher orders.

We classify the different approaches to numerical integration on cut domains in Figure 1. They can be grouped into methods relying on linear or even nonlinear approximation of either intersections or moment-fitting equations (red disks), methods that employ adaptive subdivision of the domain for the reduction to base cases or for improving the accuracy (green regions) and methods that employ lower-variate integrals or even reduce the integral to evaluations of such integrals (shown in blue). Our method avoids the computation of non-linear approximations to intersection problems and only performs a linear approximation of the boundary in each integration cell, while still achieving a high approximation order.

So far, we were able to achieve a fourth-order approximation of the integral in each cut integration cell, hence the method is suited, e.g., for Galerkin methods for second order problems that employ discretization spaces with cubic approximation order, such as piecewise quadratic polynomials. In order to extend the method to high-order quadrature, which is needed when using Galerkin methods based on high-order finite element or spline spaces, we need to be able to compute all derivatives of the integral of a function over a moving domain. In order to do so, we need a very general version of the transport theorem.

In the present work, which focuses on two-dimensional integrals, we derive a transport theorem for moving curves that may possess vertices. We then show how to apply it to the computation of *all* the derivatives of the integral over a moving domain. This result enables us to establish efficient and accurate quadrature rules for planar domains, which are cut by implicitly defined curves that are assumed to be smooth. We analyze the accuracy and the computational complexity and we provide numerical examples both for the numerical quadrature and for its application to an immersed isogeometric Galerkin method. The results of the numerical experiments indicate that the method indeed achieves high orders of convergence.

While this paper focuses on the two-dimensional case only, our presented method provides a clear path for the generalization to the three-dimensional case by extending the method proposed in [43] to higher degrees. However, the application to trimmed volumes leads to additional challenges and will be left as a direction for future work.

The rest of the paper is organized as follows: We begin in Section 2 by recapitulating the quadrature rule based on first order error correction that was introduced in [44] and [43]. Then, in Section 3, we prove the transport theorem for moving curves. Section 4 adapts the transport theorem to the special setting of implicitly defined curves and shows how to use it to compute

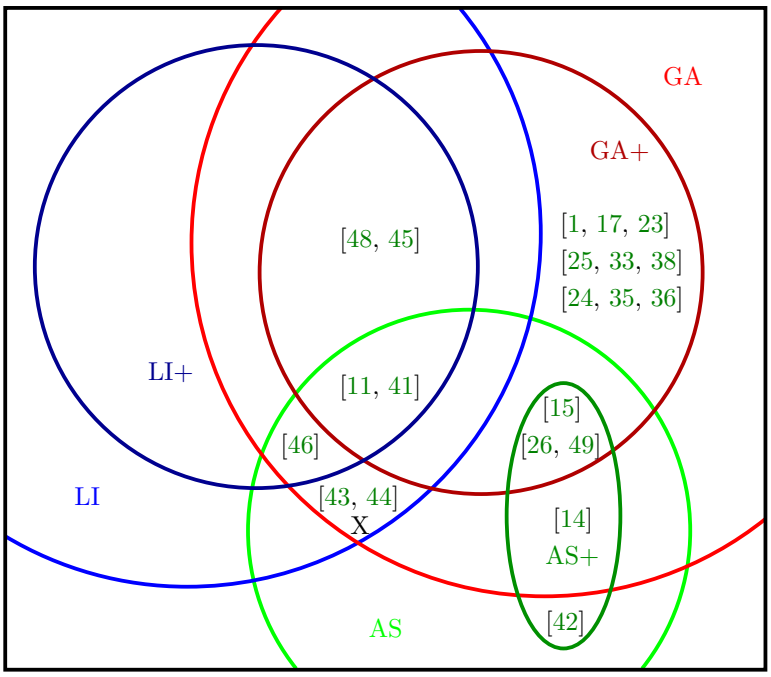


Fig. 1: Classification of different approaches for numerical integration into the categories "Linear (GA) and non-linear (GA+) geometry approximation or moment fitting" (red), "use of (LI) or even reduction to (LI+) lower-variate Integrals" (blue), and "adaptive subdivision for reduction to base cases (AS) or to increase the accuracy (AS+)" (green). The method proposed in this work is indicated by the symbol "X".

derivatives of arbitrary order in this setting. We use these results in Section 5 to define the novel high-order quadrature rule for planar domains that are cut by an implicitly defined curve, whose complexity and accuracy we then analyze in the subsequent section. The paper concludes by presenting numerical results in Section 7, which are followed by closing remarks and suggestions for future research.

2. Numerical quadrature by error correction. In the previous publication [44], a fourth-order quadrature rule for planar implicitly defined trimmed elements was presented. It is assumed that the integration region is given by the set

$$B_\tau = \{(x, y) \in B : \tau(x, y) > 0\},$$

where $B \subset \mathbb{R}^2$ is a quadrilateral (for example a trimmed element of a spline discretization) and τ is a differentiable function

$$\tau : B \to \mathbb{R},$$

called *trimming function*. The aim is to approximate the integral

$$I_\tau f = \int_{B_\tau} f(x, y)\, \mathrm{d}y\, \mathrm{d}x \tag{2.1}$$

of a function f over the computational domain.

In order to approximate the integral (2.1), the trimming function τ is first approximated by a linear function

$$\sigma(x, y) = \sigma_0 x + \sigma_1 y + \sigma_2,$$

whose function values at the vertices of B have the same signs as the original trimming function's. These signs define the *base cases* of the method. We treat zero as positive. Therefore, in the two-dimensional setting, there are two trimmed base cases: the *rectangle case* and the *triangle case*, see Figure 2. It is enforced by adaptive subdivision of B that the considered cell is in one

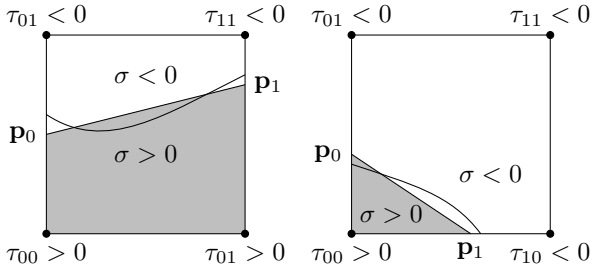


Fig. 2: The two base cases of the quadrature rule for trimmed two-dimensional domains.

of the two base cases. This is possible under the assumption that τ is smooth and does not have singular points along the curve $\tau = 0$. For details on the subdivision algorithm see also [44, Section 3].

In order to find the linear approximation σ, the intersection points $\mathbf{p}_0$ and $\mathbf{p}_1$ of the zero-level set of σ with the boundary are determined by interpolation of τ at the intersected edges' vertices. To be precise, assuming that the box B is defined by $B = [x_0, x_0 + h_x] \times [y_0, y_0 + h_y]$, we set

$$\mathbf{p}_0 = \left(x_0, y_0 + h_y \frac{\tau_{00}}{\tau_{00} - \tau_{01}}\right), \quad \mathbf{p}_1 = \left(x_0 + h_x, y_0 + h_y \frac{\tau_{10}}{\tau_{10} - \tau_{11}}\right)$$

in the rectangle case and

$$\mathbf{p}_0 = \left(x_0, y_0 + h_y \frac{\tau_{00}}{\tau_{00} - \tau_{01}}\right), \quad \mathbf{p}_1 = \left(x_0 + h_x \frac{\tau_{00}}{\tau_{00} - \tau_{10}}, y_0\right)$$

in the triangle case depicted in Figure 2.

This defines σ up to one additional degree of freedom, which is chosen by approximating a partial derivative of τ using finite differences and setting the corresponding partial derivative of σ to the same value. In the rectangle base case, $\frac{\partial \tau}{\partial y}$ is approximated while in the triangle base case the partial derivative in the direction of the longest edge component inside $\{\sigma > 0\}$ is used, see [44, Section 5] for details.

Once an approximate trimming function σ is found, the integral

$$I_\sigma f = \int_{B_\sigma} f(x, y) \, \mathrm{d}y \, \mathrm{d}x, \tag{2.2}$$

where

$$B_\sigma = \{(x, y) \in B : \sigma(x, y) > 0\}, \tag{2.3}$$

is used as an initial approximation to (2.1). Since σ is linear, B_σ is a polyhedron and the integral is therefore simple to evaluate. The resulting quadrature rule is called *Linearized Trimmed* quadrature rule (LT).

To improve the approximation, the intermediate trimming function

$$\eta(u, x, y) = \sigma(x, y) + u(\tau(x, y) - \sigma(x, y)) \tag{2.4}$$

is defined, which leads to the function of intermediate integral values

$$Q(u) = \int_{B_\eta(u)} f(x, y) \, \mathrm{d}y \, \mathrm{d}x,$$

where $B_\eta(u) = \{(x,y) \in B : \eta(u,x,y) > 0\}$. The sum of the first two terms of the Taylor expansion of Q around $u = 0$ is then used as an improved approximation

$$Q(1) \approx Q(0) + Q'(0). \tag{2.5}$$

The quadrature rule resulting from approximating the two terms on the right-hand-side of (2.5) is called *Corrected Linearized Trimmed* quadrature rule (CLT). It was shown in [44] that while the approximation (2.2) results in a third order quadrature rule (on the element level), the corrected quadrature rule resulting from (2.5) is of fourth order.

In this paper, we will extend (2.5) by adding further terms of the Taylor expansion

$$Q(1) \approx \sum_{\alpha=0}^{k} \frac{Q^{(\alpha)}(0)}{\alpha!} \tag{2.6}$$

for any $k \in \mathbb{N}_0$. We thereby define a new type of quadrature rule that consists of distinct nodes and weights for the integrand as well as for its derivatives:

$$\begin{aligned} \int_{B_\tau} f(x,y)\,\mathrm{d}y\,\mathrm{d}x &\approx \sum_{\alpha=0}^{\max(0,k-1)} \sum_{q=1}^{n_\alpha} \left\langle \omega_q^\alpha, D^\alpha f(x_q^\alpha) \right\rangle \\ &= \sum_q \omega_q^0 f(x_q^0) + \sum_q \omega_q^1 \cdot \nabla f(x_q^1) + \sum_q \left\langle \omega_q^2, H_f(x_q^2) \right\rangle_F + \ldots, \end{aligned} \tag{2.7}$$

where $x_q^\alpha \in \mathbb{R}^2$ are the quadrature nodes and $\omega_q^\alpha \in \bigotimes^\alpha \mathbb{R}^2$ are the associated quadrature weights. Here, $D^\alpha f$ is the tensor that contains all partial derivatives of order α. We conjecture that the order of this new quadrature rule is $k+3$.

The derivative $Q'(u)$ can be computed using the Reynolds transport theorem consists of an integral over the moving curve $\{\eta(u,x,y) = 0\} \cap B$ and takes the form

$$Q'(u) = -\int_{\{\eta(u,x,y)=0\}\cap B} f(x,y) \frac{\eta_u(u,x,y)}{\|\nabla \eta(u,x,y)\|}\,\mathrm{d}s. \tag{2.8}$$

It is therefore necessary to compute the derivative of this integral with respect to u in order to find the higher-order correction terms.

3. The transport theorem for moving curves. Throughout this section, let $L(u)$ be a family of oriented C^2-continuous curve segments with boundary that admits a C^1-continuous family of C^2-continuous parameterizations

$$\mathbf{q}(u,.) : [s_0(u), s_1(u)] \to L(u).$$

We choose an orientation and denote by $\mathbf{n}$ the unit normal of L, by $\mathbf{t}$ the unit tangent, by $v = \frac{\mathrm{d}q}{\mathrm{d}u}(u,s) \cdot \mathbf{n}$ the normal velocity and by κ the curvature. All these quantities depend on both the parameter $u \in \mathbb{R}$ and the point $x \in L(u)$. Additionally, we denote by $\mathbf{b}_0(u)$ the start point and by $\mathbf{b}_1(u)$ the end point of $L(u)$ with respect to the orientation, and by

$$v_{0/1}^{\mathrm{bdry}}(u) = \mathbf{t}(u, \mathbf{b}_{0/1}(u)) \cdot \frac{\mathrm{d}\mathbf{b}_{0/1}}{\mathrm{d}u}(u) \tag{3.1}$$

the tangential component of the boundary velocity.

With the motivation of computing the derivatives of (2.8), we are interested in the first derivative of the curve integral of a function F on the subcurve L with respect to the parameter u.

THEOREM 3.1 (Transport theorem for moving curves). *Given a function $F \in C^1(\mathbb{R} \times \mathbb{R}^2)$, the derivative of the curve integral over L satisfies*

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}u} \int_{L(u)} F\,\mathrm{d}s &= \int_{L(u)} \left(\frac{\partial F}{\partial u} + D_{\mathbf{n}} F v - F\kappa v \right) \mathrm{d}s \\ &\quad + F(u, \mathbf{b}_1(u)) v_1(u) - F(u, \mathbf{b}_0(u)) v_0(u), \end{aligned} \tag{3.2}$$

where $D_{\mathbf{n}}F = \nabla F \cdot \mathbf{n}$.

Proof. We consider an arbitrary $u_0 \in \mathbb{R}$ and denote by $\mathbf{q}_0(s) = \mathbf{q}(u_0, s)$ the parameterization of $L(u_0)$. We assume without loss of generality that

$$\mathbf{q}_0 : [s_0(u_0), s_1(u_0)] \to L(u_0)$$

is an arc-length parameterization. We denote by $\mathbf{n}_0(s)$ the normal field and by $v_0(s)$ the normal velocity field of $L(u_0)$.

As a consequence of Taylor's theorem there exists $\epsilon > 0$, such that for $u \in [u_0 - \epsilon, u_0 + \epsilon]$

$$\mathbf{q}(u, s) = \mathbf{q}_0(s) + (u - u_0)v_0(s)\mathbf{n}_0(s) + (u - u_0)^2\mathbf{\Gamma}(s, u). \tag{3.3}$$

Here, $\mathbf{\Gamma}(s, u)$ is a bounded remainder term.

We rewrite the integral of F over $L(u)$ as

$$\int_{L(u)} F(u, x)\,\mathrm{d}s = \int_{s_0(u)}^{s_1(u)} F(u, \mathbf{q}(u, s)) \left\| \frac{\partial \mathbf{q}}{\partial s}(u, s) \right\| \mathrm{d}s$$

and its derivative with respect to u as

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}u} \int_{L(u)} F\,\mathrm{d}s = \int_{s_0(u)}^{s_1(u)} \left(\frac{\partial F}{\partial u} + \nabla F \cdot \frac{\partial \mathbf{q}}{\partial u} \right) \left\| \frac{\partial \mathbf{q}}{\partial s} \right\| + F \frac{\partial}{\partial u} \left\| \frac{\partial \mathbf{q}}{\partial s} \right\| \mathrm{d}s \\ + \frac{\mathrm{d}s_1}{\mathrm{d}u}(u) F(u, \mathbf{q}(u, s_1(u))) \left\| \frac{\partial \mathbf{q}}{\partial s}(u, s_1(u)) \right\| \\ - \frac{\mathrm{d}s_0}{\mathrm{d}u}(u) F(u, \mathbf{q}(u, s_0(u))) \left\| \frac{\partial \mathbf{q}}{\partial s}(u, t_0(u)) \right\|. \end{aligned} \tag{3.4}$$

According to (3.3), the appearing derivatives of $\mathbf{q}$ are

$$\frac{\partial \mathbf{q}}{\partial u}(u, s) = v_0(s)\mathbf{n}_0(s) + 2(u - u_0)\mathbf{\Gamma}(u, s) + (u - u_0)^2 \frac{\partial \mathbf{\Gamma}}{\partial u}(u, s), \tag{3.5}$$

$$\frac{\partial \mathbf{q}}{\partial s}(u, s) = \frac{\partial \mathbf{q}_0}{\partial s}(s) + (u - u_0) \left(\frac{\partial v_0}{\partial s}(s)\mathbf{n}_0(s) + v_0(s) \frac{\partial \mathbf{n}_0}{\partial s}(s) \right) + (u - u_0)^2 \frac{\partial \mathbf{\Gamma}}{\partial s}(u, s) \tag{3.6}$$

and

$$\frac{\partial^2 \mathbf{q}}{\partial s \partial u}(u, s) = \frac{\partial v_0}{\partial s}(s)\mathbf{n}_0(s) + v_0(s) \frac{\partial \mathbf{n}_0}{\partial s}(s) + 2(u - u_0) \frac{\partial \mathbf{\Gamma}}{\partial s}(u, s) + (u - u_0)^2 \frac{\partial^2 \mathbf{\Gamma}}{\partial u \partial s}(u, s). \tag{3.7}$$

The norm of the tangent evolves as

$$\frac{\partial}{\partial u} \left(\left\| \frac{\partial \mathbf{q}}{\partial s} \right\| \right) = \frac{\frac{\partial \mathbf{q}}{\partial s} \cdot \frac{\partial^2 \mathbf{q}}{\partial s \partial u}}{\| \frac{\partial \mathbf{q}}{\partial s} \|}$$

and we therefore have at $u = u_0$

$$\begin{aligned} \frac{\partial}{\partial u} \left(\left\| \frac{\partial \mathbf{q}}{\partial s} \right\| \right)(u_0, s) &= \frac{\frac{\partial \mathbf{q}_0}{\partial s}(s)}{\| \frac{\partial \mathbf{q}_0}{\partial s}(s) \|} \cdot \left(\frac{\partial v_0}{\partial s}(s)\mathbf{n}_0(s) + v_0(s) \frac{\partial \mathbf{n}_0}{\partial s}(s) \right) \\ &= \frac{\partial \mathbf{q}_0}{\partial s}(s) \cdot v_0(s) \frac{\partial \mathbf{n}_0}{\partial s}(s) \\ &= -\kappa_0(s) v_0(s). \end{aligned} \tag{3.8}$$

We substitute (3.5)-(3.8) into (3.4) and evaluate at $u = u_0$, arriving at

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}u}\int_{L(u)} F\,\mathrm{d}s\bigg|_{u=u_0} &= \int_{s_0(u_0)}^{s_1(u_0)} \left(\frac{\partial F}{\partial u} + \nabla F \cdot \frac{\partial \mathbf{q}}{\partial u}\bigg|_{u=u_0}\right) \left\|\frac{\partial \mathbf{q}_0}{\partial s}\right\| - F\kappa_0(s)v_0(s)\,\mathrm{d}s \\
&\quad + \frac{\mathrm{d}s_1}{\mathrm{d}u}(u_0)F(u_0, \mathbf{q}(u_0, s_1(u_0)))\left\|\frac{\partial \mathbf{q}_0}{\partial s}(u_0, s_1(u_0))\right\| \\
&\quad - \frac{\mathrm{d}s_0}{\mathrm{d}u}(u_0)F(u_0, \mathbf{q}(u_0, s_0(u_0)))\left\|\frac{\partial \mathbf{q}_0}{\partial s}(u_0, s_0(u_0))\right\| \\
&= \int_{s_0(u_0)}^{s_1(u_0)} \frac{\partial F}{\partial u} + \nabla F \cdot \frac{\partial \mathbf{q}}{\partial u}\bigg|_{u=u_0} - F\kappa_0(s)v_0(s)\,\mathrm{d}s \\
&\quad + \frac{\mathrm{d}s_1}{\mathrm{d}u}(u_0)F(u_0, \mathbf{q}(u_0, s_1(u_0))) - \frac{\mathrm{d}s_0}{\mathrm{d}u}(u_0)F(u_0, \mathbf{q}(u_0, s_0(u_0))).
\end{aligned} \tag{3.9}$$

Using the definition of $\mathbf{q}$, the boundary points are defined as

$$\mathbf{b}_{0/1}(u) = \mathbf{q}(u, s_{0/1}(u))$$

and their derivatives satisfy

$$\frac{\mathrm{d}\mathbf{b}_{0/1}}{\mathrm{d}u} = \frac{\partial \mathbf{q}}{\partial u} + \frac{\partial \mathbf{q}}{\partial s}\frac{\mathrm{d}s_{0/1}}{\mathrm{d}u}.$$

Consequently, the tangential part of the velocity fulfills

$$v_{0/1}^{\mathrm{bdry}} = \frac{\mathrm{d}\mathbf{b}_{0/1}}{\mathrm{d}u} \cdot \mathbf{t} = \frac{\partial \mathbf{q}}{\partial u} \cdot \frac{\frac{\partial \mathbf{q}}{\partial s}}{\|\frac{\partial \mathbf{q}}{\partial s}\|} + \left\|\frac{\partial \mathbf{q}}{\partial s}\right\| \frac{\mathrm{d}s_{0/1}}{\mathrm{d}u} = \left\|\frac{\partial \mathbf{q}}{\partial s}\right\| \frac{\mathrm{d}s_{0/1}}{\mathrm{d}u}. \tag{3.10}$$

Using (3.5), (3.9), (3.10) and the fact that u_0 was chosen arbitrarily completes the proof. ∎

Remark 3.2 (Higher derivatives). On a first glance it seems as if the computation of higher derivatives was possible by simply applying Theorem 3.1 repeatedly. However, in general the integrand on the right-hand-side of (3.2) is itself not a function in $C^1(\mathbb{R} \times \mathbb{R}^2)$ and Theorem 3.1 cannot be applied to it directly. In the next Section, we will apply the transport theorem to implicitly defined moving curves. We will see that in this special case, the transport theorem can be applied repeatedly and can therefore be used to compute all derivatives with respect to u.

Remark 3.3. The name *transport theorem* for a theorem regarding the differentiation of an integral over a domain that varies in time stems from the applications of the Reynolds transport theorem in continuum mechanics.

4. Implicit curves intersecting a fixed domain. We apply Theorem 3.1 to the special case of implicitly defined curves that intersect a fixed domain. Our goal is to derive easy-to-evaluate expressions for all derivatives of the integral over an implicitly defined moving curve.

4.1. Definitions. Let $\eta \in C^{1,2}(\mathbb{R} \times \mathbb{R}^2)$ be such that its gradient with respect to the space components along the zero-level sets

$$\eta(u, x, y) = 0 \tag{4.1}$$

satisfies $\nabla\eta \neq 0$. The zero-level sets (that may possess more than one connected component) then form a family of curves that we denote by $A(u)$.

Furthermore, we assume that the curve segment $L(u)$ is the intersection

$$L(u) = A(u) \cap \bar{B} \tag{4.2}$$

of the implicit curve $A(u)$ with the closure of a bounded region $B \subset \mathbb{R}^2$. More precisely, we assume that B is chosen such that the intersection consists of *exactly one curve segment*, in order to keep

the presentation simple. The case of multiple segments can be dealt with by considering them individually. Other situations – such as closed curves contained in B – are not considered.

Since $A(u)$ is defined implicitly in (4.1), we have the following representations of the normal, normal velocity, tangent and curvature for one choice of orientation (see [19]):

$$\mathbf{n} = \frac{-\nabla\eta}{\|\nabla\eta\|}, \quad v = \frac{\eta_u}{\|\nabla\eta\|}, \quad \mathbf{t} = \frac{1}{\|\nabla\eta\|}\nabla\eta^\perp \tag{4.3}$$

and

$$\kappa = \frac{\mathbf{t}\cdot H_\eta \mathbf{t}}{\|\nabla\eta\|} = \frac{\nabla\eta^\perp \cdot H_\eta \nabla\eta^\perp}{\|\nabla\eta\|^3}, \tag{4.4}$$

where

$$H_\eta = \begin{pmatrix} \eta_{xx} & \eta_{xy} \\ \eta_{xy} & \eta_{yy} \end{pmatrix}$$

is the Hessian and

$$\nabla\eta^\perp = \begin{pmatrix} -\eta_y \\ \eta_x \end{pmatrix}$$

is the gradient rotated by $\frac{\pi}{2}$ counterclockwise. Clearly, all of these quantities depend on u, x, and y. Strictly speaking, they are only defined if $\eta(u,x,y) = 0$.

Additionally, we assume that the region B is defined implicitly by a function $\psi \in C(R^2)$, such that

$$B = \{\psi(x,y) > 0\}.$$

For points $\mathbf{p} \in \partial B$ that satisfy $\nabla\psi(\mathbf{p}) \neq 0$, we denote by

$$\boldsymbol{\nu} = \frac{-\nabla\psi}{\|\nabla\psi\|}$$

the outwards-pointing normal of the boundary ∂B. Furthermore, we define $\mathbf{b}_0(u)$ to be the start point, and $\mathbf{b}_1(u)$ to be the end point of $L(u)$ on the boundary $\{\eta = 0\} \cap \partial B$, with respect to the orientation defined by the tangent $\mathbf{t}$.

4.2. The transport theorem for implicit curves. In order to express the derivative of the integral over an implicit moving curve, we first need to find a representation of the tangential part of velocity of its boundary.

LEMMA 4.1. *The tangential part of the velocity of the boundary of $L(u)$ at the start or end point $\mathbf{b}_{0/1}(u)$ is given by*

$$v_{0/1}^{bdry}(u) = \left.\frac{-\eta_u \boldsymbol{\nu}\cdot\nabla\eta}{\nabla\eta\cdot\boldsymbol{\nu}^\perp\|\nabla\eta\|}\right|_{(u,\mathbf{b}_{0/1}(u))}. \tag{4.5}$$

Proof. Because $\mathbf{b}_{0/1}(u)$ lies on ∂B for all u, we have

$$\frac{\partial \mathbf{b}_{0/1}}{\partial u}(u) = \lambda(u)\boldsymbol{\nu}^\perp(\mathbf{b}_{0/1}(u)) \tag{4.6}$$

for some function $\lambda(u)$. Since $\mathbf{b}_{0/1}(u)$ also lies on $A(u)$, it also fulfills $\eta(u, \mathbf{b}_{0/1}(u)) = 0$ and therefore

$$\frac{\partial\eta}{\partial u} + \nabla\eta\cdot\lambda(u)\boldsymbol{\nu}^\perp = 0.$$

This implies

$$\lambda(u) = \left.\frac{-\eta_u}{\nabla\eta\cdot\boldsymbol{\nu}^\perp}\right|_{(u,\mathbf{b}_{0/1}(u))}.$$

By substituting λ into (4.6) and using the definitions (3.1) and (4.3) we conclude (4.5). ∎

With this representation for the tangential part of the boundary velocity we can state the *transport theorem for moving implicit curves*:

THEOREM 4.2. *We assume that the number of boundary intersections* $\mathbf{b} \in \{\eta = 0\} \cap \partial B$ *is finite and that for all* u *the boundary intersections satisfy* $\nabla\psi(\mathbf{b}) \neq 0$*. Then, the derivative of the integral of a function* $F \in C^1(\mathbb{R} \times \mathbb{R}^2)$ *over the moving curve satisfies*

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}u}\int_{\{\eta=0\}\cap B} F\,\mathrm{d}s = &\int_{\{\eta=0\}\cap B}\left(\frac{\partial F}{\partial u} + \nabla F\cdot\frac{-\eta_u\nabla\eta}{\|\nabla\eta\|^2} - F\frac{\eta_u\nabla\eta^\perp\cdot H_\eta\nabla\eta^\perp}{\|\nabla\eta\|^4}\right)\mathrm{d}s\\ &+\sum_{\mathbf{b}\in\{\eta=0\}\cap\partial B} F\frac{\eta_u\boldsymbol{\nu}\cdot\nabla\eta}{|\nabla\eta\cdot\boldsymbol{\nu}^\perp|\,\|\nabla\eta\|}\bigg|_{(u,\mathbf{b}(u))}.\end{aligned} \tag{4.7}$$

Proof. The equation follows from Theorem 3.1, Lemma 4.1 and (4.3)-(4.4). In particular, the signs of the boundary terms are correctly taken into account since at the start point $\mathbf{b}_0$ of each curve segment we have

$$\operatorname{sign}(\nabla\eta(u, b_0(u))\cdot\boldsymbol{\nu}^\perp(\mathbf{b}_0(u))) = 1$$

while at the end point $\mathbf{b}_1$ of each curve segment we have

$$\operatorname{sign}(\nabla\eta(u, b_1(u))\cdot\boldsymbol{\nu}^\perp(\mathbf{b}_1(u))) = -1.$$ ∎

4.3. Higher order derivatives. Assuming sufficient smoothness of η, ψ and F, we observe that the right-hand side in Theorem 4.2 consists of an integral of a function in $C^1(\mathbb{R}\times\mathbb{R}^2)$ over the moving curve and of point evaluations of another $C^1(\mathbb{R}\times\mathbb{R}^2)$-function at the boundary intersections.

For simplicity, we once more consider there to be single component of $\eta = 0$ inside B. By setting

$$\begin{aligned}F^{(1)} &= \frac{\partial F}{\partial u} + D_n F v - F\kappa v\\ &= \frac{\partial F}{\partial u} + \nabla F\cdot\frac{-\eta_u\nabla\eta}{\|\nabla\eta\|^2} - F\frac{\eta_u\nabla\eta^\perp\cdot H_\eta\nabla\eta^\perp}{\|\nabla\eta\|^4}\end{aligned}$$

and

$$\begin{aligned}B_{0/1}^{(1)}(u) &= F(u,\mathbf{b}_{0/1}(u))v_{0/1}^{\mathrm{bdry}}(u),\\ &= F\frac{\eta_u\boldsymbol{\nu}\cdot\nabla\eta}{|\nabla\eta\cdot\boldsymbol{\nu}^\perp|\,\|\nabla\eta\|}\bigg|_{(u,\mathbf{b}_{0/1}(u))},\end{aligned}$$

we can write the first derivative as

$$\frac{\mathrm{d}}{\mathrm{d}u}\int_{L(u)} F\,\mathrm{d}s = \int_{L(u)} F^{(1)}\,\mathrm{d}s + B_1^{(1)}(u) - B_0^{(1)}(u).$$

In order to compute the second order derivative, we therefore need to compute the derivative of an integral term as well as the derivatives of the boundary terms. The derivative of the integral can be evaluated by another application of Theorem 4.2. In order to compute the derivatives of the boundary terms we observe that, the chain rule implies

$$\frac{\mathrm{d}}{\mathrm{d}u}G(u,\mathbf{b}_{0/1}(u)) = \frac{\partial G}{\partial u}(u,\mathbf{b}_{0/1}(u)) + \nabla G(u,\mathbf{b}_{0/1}(u))\cdot\frac{\mathrm{d}\mathbf{b}_{0/1}}{\mathrm{d}u}(u). \tag{4.8}$$

for any differentiable function G.

Using these representations, we can express the second derivative of the curve integral,

$$\begin{aligned}\frac{\mathrm{d}^2}{\mathrm{d}u^2}\int_{L(u)} F\,\mathrm{d}s = \int_{L(u)} &\frac{\partial F^{(1)}}{\partial u} + D_n F^{(1)} v - F^{(1)}\kappa v\,\mathrm{d}s \\ &+ F^{(1)}(u,\mathbf{b}_1(u))v_1(u) - F^{(1)}(u,\mathbf{b}_0(u))v_0(u) \\ &+ \frac{\partial B_1^{(1)}}{\partial u}(u,\mathbf{b}_1(u)) + \nabla B_1^{(1)}(u,\mathbf{b}_1(u))\cdot\frac{\mathrm{d}\mathbf{b}_1}{\mathrm{d}u}(u) \\ &- \frac{\partial B_0^{(1)}}{\partial u}(u,\mathbf{b}_0(u)) - \nabla B_0^{(1)}(u,\mathbf{b}_0(u))\cdot\frac{\mathrm{d}\mathbf{b}_0}{\mathrm{d}u}(u).\end{aligned}$$

Consequently, we can use Theorem 4.2 to compute all derivatives of the curve integral, also in the case of multiple components of the curve inside B.

We define the two linear operators $\mathrm{U} : C^k(\mathbb{R}\times\mathbb{R}^2) \to C^{k-1}(\mathbb{R}\times\mathbb{R}^2)$,

$$\begin{aligned}\mathrm{U}G &= \frac{\partial G}{\partial u} + D_n G v - G\kappa v \\ &= \frac{\partial G}{\partial u} + \nabla G\cdot\frac{-\eta_u\nabla\eta}{\|\nabla\eta\|^2} - G\frac{\eta_u\nabla\eta^\perp\cdot H_\eta\nabla\eta^\perp}{\|\nabla\eta\|^4}\end{aligned}$$

and $\mathrm{V} : C^0(\mathbb{R}\times\mathbb{R}^2) \to C^0(\mathbb{R})$,

$$\mathrm{V}G(u) = \sum_{\mathbf{b}\in\{\eta=0\}\cap\partial B} G\frac{\eta_u\boldsymbol{\nu}\cdot\nabla\eta}{|\nabla\eta\cdot\boldsymbol{\nu}^\perp|\,\|\nabla\eta\|}\bigg|_{(u,\mathbf{b}(u))}$$

and obtain

THEOREM 4.3. *Given a function $F \in C^k(\mathbb{R}\times\mathbb{R}^2)$, the derivatives of the curve integral over L satisfy*

$$\frac{\mathrm{d}^k}{\mathrm{d}u^k}\int_{L(u)} F\,\mathrm{d}s = \int_{L(u)} \mathrm{U}^k F\,\mathrm{d}s + \sum_{\substack{i+j=k-1\\ i,j\geq 0}} \frac{\mathrm{d}^i}{\mathrm{d}u^i}\mathrm{V}\mathrm{U}^j F. \tag{4.9}$$

Proof. The proof follows by repeatedly applying Theorem 4.2. ∎

Finally, we need to evaluate the differential operator $\frac{\mathrm{d}^i}{\mathrm{d}u^i}$ in Theorem 4.3 explicitly.

LEMMA 4.4. *Given a function $G \in C^1(\mathbb{R}\times\mathbb{R}^2)$ we have*

$$\frac{\mathrm{d}}{\mathrm{d}u}G(u,\mathbf{b}_{0/1}(u)) = \left(\frac{\partial G}{\partial u} - \nabla G\cdot\left(\frac{\eta_u}{\|\nabla\eta\|^2}\nabla\eta - \frac{\eta_u\boldsymbol{\nu}\cdot\nabla\eta}{\nabla\eta\cdot\boldsymbol{\nu}^\perp\|\nabla\eta\|^2}\nabla\eta^\perp\right)\right)\bigg|_{(u,\mathbf{b}_{0/1}(u))}. \tag{4.10}$$

Proof. The proof follows from (4.8), Lemma 4.1 and (4.3)-(4.4), ∎

5. Error correction terms of the 2D corrected trimmed quadrature rule. We use Theorem 4.3 to compute all correction terms of the approximation of the integral over a trimmed domains presented in [44] and [43]. Analogously to the first order error correction that we summarized in section 2, we need to evaluate the terms of Taylor expansion

$$Q(1) \approx Q(0) + Q'(0) + \frac{Q''(0)}{2} + \frac{Q'''(0)}{6}\ldots + \frac{Q^{(k)}(0)}{k!} \tag{5.1}$$

of the intermediate integral values

$$Q(u) = \int_{B_\eta(u)} f(x,y)\,\mathrm{d}y\,\mathrm{d}x,$$

where $B_\eta(u) = \{(x,y) \in B : \eta(u,x,y) > 0\}$ and

$$\eta(u,x,y) = \sigma(x,y) + u(\tau(x,y) - \sigma(x,y)).$$

In [43, Lemma 1] it was shown that the first derivative of the integral over the implicitly defined moving domain is given by

$$Q'(u) = -\int_{\{\eta=0\}\cap B} f \frac{\eta_u}{\|\nabla\eta\|}\,\mathrm{d}s \tag{5.2}$$

and that the first order correction term therefore satisfies

$$Q'(0) = -\int_{\{\sigma=0\}\cap B} f \frac{\tau}{\|\nabla\sigma\|}\,\mathrm{d}s.$$

We observe that (5.2) is an integral of the function

$$F(u,x,y) = f(x,y)\frac{\eta_u(u,x,y)}{\|\nabla\eta(u,x,y)\|}$$

over a moving family of curves that is defined as the intersection of implicitly defined curves $\{\eta = 0\}$ and a fixed region B. We first use Theorem 4.2 to compute the second order correction term.

Corollary 5.1 (Second order correction term). *The second order correction term takes the form*

$$Q''(0) = -\int_{\{\sigma=0\}\cap B} \frac{\left(\tau^2\nabla f + 2f\tau\left(\nabla\tau - \nabla\sigma\right)\right)\cdot\nabla\sigma}{\|\nabla\sigma\|^3}\,\mathrm{d}s + C,$$

where

$$C = \frac{f(\mathbf{p}_1)\tau(\mathbf{p}_1)^2}{\frac{\sigma_x^3}{\sigma_y} + \sigma_x\sigma_y} + \frac{f(\mathbf{p}_0)\tau(\mathbf{p}_0)^2}{\frac{\sigma_y^3}{\sigma_x} + \sigma_x\sigma_y} \tag{5.3}$$

if B is in the triangle base case and

$$C = -\frac{f(\mathbf{p}_1)\tau(\mathbf{p}_1)^2}{\frac{\sigma_y^3}{\sigma_x} + \sigma_x\sigma_y} + \frac{f(\mathbf{p}_0)\tau(\mathbf{p}_0)^2}{\frac{\sigma_y^3}{\sigma_x} + \sigma_x\sigma_y} \tag{5.4}$$

if B is in the rectangle base case (see Figure 2). Here, $p_0 = \mathbf{b}_0(0)$ and $p_1 = \mathbf{b}_1(0)$ are the start and end points of $L(0)$.

Proof. In order to compute the derivative of (5.2) we set

$$F = f\frac{\eta_u}{\|\nabla\eta\|}.$$

The derivative with respect to u is given by

$$F_u = f\frac{\eta_{uu}\|\nabla\eta\| - \eta_u\frac{\nabla\eta_u\cdot\nabla\eta}{\|\nabla\eta\|}}{\|\nabla\eta\|^2} = -f\frac{\eta_u\nabla\eta_u\cdot\nabla\eta}{\|\nabla\eta\|^3}, \tag{5.5}$$

where we used the definition of η from (2.4) and the assumption that σ is linear to see that $\eta_{uu} = 0$. Computing the gradient of F, we arrive at

$$\nabla F = f\frac{\nabla\eta_u\|\nabla\eta\| - \eta_u\frac{H_\eta\nabla\eta}{\|\nabla\eta\|}}{\|\nabla\eta\|^2} + \nabla f\frac{\eta_u}{\|\nabla\eta\|}. \tag{5.6}$$

From (2.4), (5.5) and (5.6), we infer

$$F_u|_{u=0} = -f\frac{(\tau-\sigma)(\nabla\tau-\nabla\sigma)\cdot\nabla\sigma}{\|\nabla\sigma\|^3} \tag{5.7}$$

and

$$\nabla F|_{u=0} = f\frac{(\nabla\tau-\nabla\sigma)}{\|\nabla\sigma\|} + \nabla f\frac{(\tau-\sigma)}{\|\nabla\sigma\|}. \tag{5.8}$$

Using Theorem 4.2 and the observation that $D^2\eta = 0$, we obtain

$$\begin{aligned} Q''(0) = &\int_{\{\sigma=0\}\cap B} -f\frac{(\tau-\sigma)(\nabla\tau-\nabla\sigma)\cdot\nabla\sigma}{\|\nabla\sigma\|^3}\,\mathrm{d}s \\ &+ \int_{\{\sigma=0\}\cap B}\left(f\frac{(\nabla\tau-\nabla\sigma)}{\|\nabla\sigma\|} + \nabla f\frac{(\tau-\sigma)}{\|\nabla\sigma\|}\right)\cdot\frac{-(\tau-\sigma)\nabla\sigma}{\|\nabla\sigma\|^2}\,\mathrm{d}s \\ &+ F\frac{-(\tau-\sigma)\boldsymbol{\nu}\cdot\nabla\sigma}{\nabla\sigma\cdot\boldsymbol{\nu}^\perp\|\nabla\sigma\|}\bigg|_{(u,\mathbf{b}_1(u))} - F\frac{-(\tau-\sigma)\boldsymbol{\nu}\cdot\nabla\sigma}{\nabla\sigma\cdot\boldsymbol{\nu}^\perp\|\nabla\sigma\|}\bigg|_{(u,\mathbf{b}_0(u))} \\ = &-\int_{\{\sigma=0\}\cap B}\frac{\tau^2\nabla f\cdot\nabla\sigma + 2f\tau\,(\nabla\tau-\nabla\sigma)\cdot\nabla\sigma}{\|\nabla\sigma\|^3}\,\mathrm{d}s \\ &+ F\frac{-\tau\boldsymbol{\nu}\cdot\nabla\sigma}{\nabla\sigma\cdot\boldsymbol{\nu}^\perp\|\nabla\sigma\|}\bigg|_{(u,\mathbf{b}_1(u))} - F\frac{-\tau\boldsymbol{\nu}\cdot\nabla\sigma}{\nabla\sigma\cdot\boldsymbol{\nu}^\perp\|\nabla\sigma\|}\bigg|_{(u,\mathbf{b}_0(u))}. \end{aligned} \tag{5.9}$$

(5.10)

Note, that $\sigma = 0$ in the integration region.

In the triangle base case, the outward-pointing normals of the boundary are

$$\boldsymbol{\nu}^0 = \boldsymbol{\nu}(u,\mathbf{b}_0(u)) = \begin{pmatrix}-1\\0\end{pmatrix} \quad\text{and}\quad \boldsymbol{\nu}^1 = \boldsymbol{\nu}(u,\mathbf{b}_1(u)) = \begin{pmatrix}0\\-1\end{pmatrix},$$

while in the rectangle base case they are $\boldsymbol{\nu}^0 = \begin{pmatrix}-1\\0\end{pmatrix}$ and $\boldsymbol{\nu}^1 = \begin{pmatrix}1\\0\end{pmatrix}$. Substituting these values into the two last terms in (5.10) leads to (5.3)-(5.4). ∎

We can use Theorem 4.3 to compute all further correction terms. Because of the definition of the operators $\mathrm{V}_{0/1}$ and U in Theorem 4.3 and the representation of the boundary derivatives in Lemma 4.4, each additional error correction term contains an additional derivative of the integrand f. If we approximate all appearing curve integrals using a standard quadrature rule, such as Gaussian quadrature, this leads to a new quadrature rule in the form of (2.7). We call it *k-times corrected linearized trimmed quadrature rule* and denote it by kCLT, where k is the number of correction terms.

Remark 5.2 (Extension to the three-dimensional case). In [8, (5.6)], a transport theorem for surface integrals is presented. The derivative of the integral over a moving surface the transport consists of the sum of another integral over the moving surface and an integral over the moving boundary curves, which in our application are the intersections of the implicitly defined surface with the faces of the box. This means that together with Theorem 4.3, all correction terms can be computed also in the three-dimensional case, thereby extending the method that was proposed in [43].

6. Computational complexity and accuracy. In this analysis and in the following numerical experiments, we approximate the univariate integrals that appear in the correction terms using standard univariate Gaussian quadrature. For the bivariate integrals we use tensor-product Gauss rules. Certainly, these quadrature rules can be replaced by any other standard quadrature rule that is suitable for polygonal domains and achieves the appropriate order.

We expect that the error of kCLT converges with order $k+3$ on a single cell with respect to the cell size, assuming that the appearing integrals are approximated well enough. Therefore, we

need to choose quadrature rules for the appearing univariate and bivariate integrals that match this approximation order. Because Gaussian quadrature with q nodes is exact for polynomials of degree $2q-1$, whose approximation order is $2q$, this means that we need to use

$$q = \left\lceil \frac{k+2}{2} \right\rceil$$

nodes per direction for both the univariate and the bivariate quadrature rules.

More precisely, we consider the compound quadrature rule

$$\int_{\Omega_\tau} f(x,y)\,\mathrm{d}y\,\mathrm{d}x \approx \sum_{B_\tau \in Q_{\text{trimmed}}} k\text{CLT}(B_\tau, f) + \sum_{B \in Q_{\text{untrimmed}}} \text{Gauss}(B, f), \tag{6.1}$$

where $\text{Gauss}(B, f)$ is the result of approximating the integral of f over B using a bivariate tensor-product Gauss rule, over a subdivision

$$Q_{\text{trimmed}} \cup Q_{\text{untrimmed}}$$

of

$$\Omega_\tau = \{(x,y) \in \Omega : \tau(x,y) \geq 0\}$$

into trimmed and untrimmed cells of uniform size h.

We assume that there are asymptotically

$$N = h^{-2}$$

untrimmed cells and

$$n = h^{-1}$$

trimmed cells in the subdivision. In order to achieve an order of convergence of $k+2$ for the quadrature error, we therefore need to choose a quadrature rule of order $k+3$ for the trimmed cells and a quadrature rule of order $k+4$ for the untrimmed cells. This can be achieved by choosing kCLT with

$$q_{\text{trimmed}} = \left\lceil \frac{k+2}{2} \right\rceil$$

Gauss nodes per direction for the trimmed cells and a tensor-product Gaussian quadrature rule with

$$q_{\text{untrimmed}} = \left\lceil \frac{k+3}{2} \right\rceil$$

nodes per direction for the untrimmed cells.

THEOREM 6.1. *In terms of function evaluations of the integrand and its derivatives, the complexity of the compound rule (6.1) based on kCLT is*

$$O\left(Nk^2 + n\left(k^2 + 2^k k + 2^k\right)\right)$$

Proof. On every trimmed cell in Q_{trimmed}, we evaluate the integrand at the quadrature nodes belonging to the bivariate quadrature rule for the approximated integration region $\{\sigma > 0\} \cap B$. Moreover, we evaluate the first $k-1$ derivatives of the integrand at the nodes corresponding to the univariate quadrature rule for the line integral over $\{\sigma = 0\} \cap B$. Finally, we need to evaluate the first $k-2$ at the start and end points of the line. Taking into account the dimension of the k-th derivative of a function, this means that we perform

$$q_{\text{trimmed}}^2 + \sum_{\alpha=0}^{\max(0,k-1)} 2^\alpha q_{\text{trimmed}} + 2 \sum_{\alpha=0}^{\max(0,k-2)} 2^\alpha$$

function evaluations on each trimmed cell. Because the number of quadrature nodes for both the trimmed and untrimmed cells depend linearly on k, the result follows. ∎

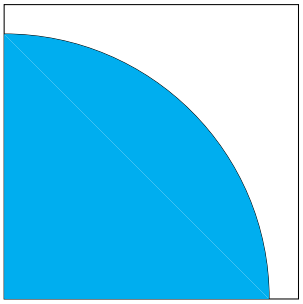


Fig. 3: The integration region

Remark 6.2. When considering the dependence on h, we observe that the estimate in Theorem 6.1 is dominated by the first term, which corresponds to the bivariate Gauss quadrature on the untrimmed cells. In particular, this means that the computational complexity of kCLT is asymptotically the same as the one of classical Gauss quadrature, while achieving a much higher accuracy. Furthermore, the asymptotic complexity with respect to h is independent of the number of error correction terms, which makes high-order kCLT very efficient.

Remark 6.3. Using the same size for untrimmed and trimmed cells in this analysis is motivated by the application of the method to immersed boundary method and trimmed isogeometric analysis, where the mesh size is given by the discrete space.

7. Numerical examples. We will now demonstrate the accuracy and efficiency of our approach in a number of numerical experiments. We implemented the method using the open source C++ library G+Smo [30]. For the approximation of the bivariate and univariate integrals appearing in the correction terms, we used standard Gauss quadrature with an appropriate number of nodes.

7.1. Area approximation. In our first example, we compute the area of a quarter circle of radius 0.9 inscribed in the unit square $B = [0, 1]^2$. It is defined by the trimming function

$$\tau(x, y) = -x^2 - y^2 + 0.81,$$

see Figure 3. The integrand is set to $f(x, y) = 1$.

We compare quadrature rules from zero to three error correction terms, denoted by LT (linearized trimmed quadrature rules) and kCLT (k-times corrected linearized trimmed quadrature rule). Figure 4 confirms that each additional error correction term results in an additional order of convergence of the quadrature error.

7.2. Integrating a function over a trimmed domain. In our next example, we approximate the integral of the function

$$f(x, y) = 32x^6y - 48x^4y^2 + 18x^2y^3 - 1 \tag{7.1}$$

over a disk of radius 0.3 inscribed in the unit square, meaning that the trimming function is

$$\tau(x, y) = (x - 0.5)^2 + (y - 0.5)^2 - 0.3^2.$$

The value of the integral can be evaluated exactly and is equal to $\frac{-7526007\pi}{100000000}$.

Figure 5 shows the integrand that we consider in this example. The main difference in this example compared to our last example, is that the derivatives of the integrand are non-zero and therefore the related terms of the quadrature rule (2.7) appear in the computation. As in our last example, we compare the convergence rates of the quadrature error when using quadrature rules with an increasing number of error correction terms. In Figure 6 we observe that adding more error correction terms results in higher orders of convergence also in this example.

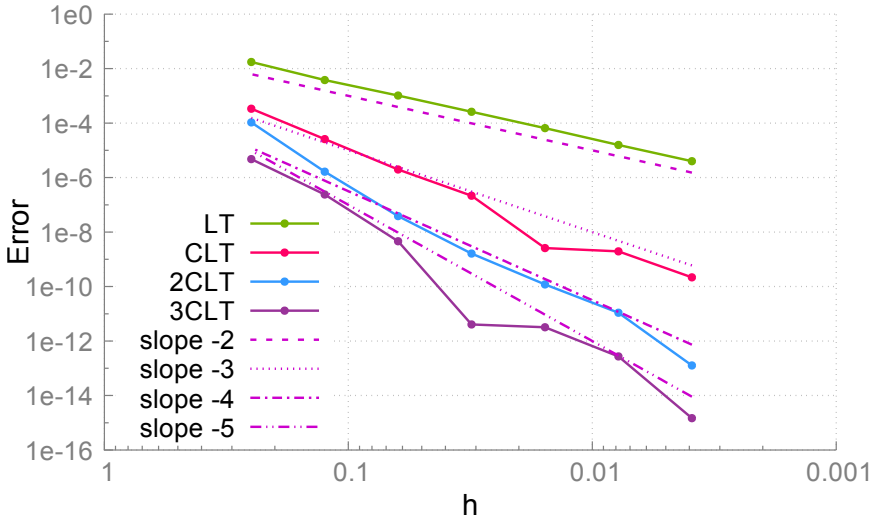


Fig. 4: Error convergence when approximating the area of a quarter circle with increasing number of error correction terms

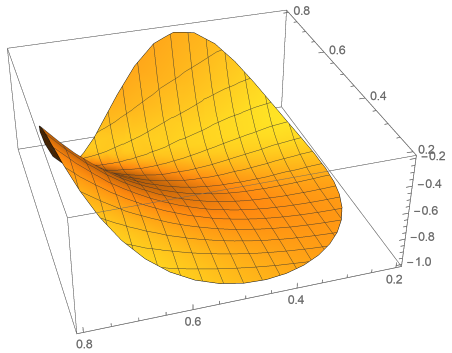


Fig. 5: The integrand defined in (7.1) over a disk of radius $r = 0.3$.

Next, we will investigate the computational complexity of our method. To this end, we record the computation times needed for achieving the approximations presented in Figure 6. In order to make a fair comparison, we use the minimum number of Gauss nodes for the univariate and bivariate Gauss quadrature rules that are employed for approximating the appearing integrals in all corrected linearized trimmed quadrature rules. This means that for LT we use 1 node per direction, for CLT and 2CLT we use 2 nodes per direction and for 3CLT we use 3 nodes per direction. Figure 7 shows the computation times. We observe that asymptotically the approximation is computed in quadratic time for all methods, which is also the same asymptotic time needed for classical Gauss quadrature. This confirms the results of Theorem 6.1 and Remark 6.2. Additionally, the overhead cost of each additional error correction term is very small compared to the increase in accuracy that we observed in Figure 7.

The assumptions in Theorem 4.2 exclude the case that the trimming curves passes exactly through a vertex of the cell, a case which is numerically very unlikely and is dealt with by the choice to treat zero values at the vertices as positive for determining the base cases in section 2. Moreover, the corresponding boundary term as well as its derivatives vanish in this case since τ and σ coincide. In order to analyze the behavior of the method when the intersection points is

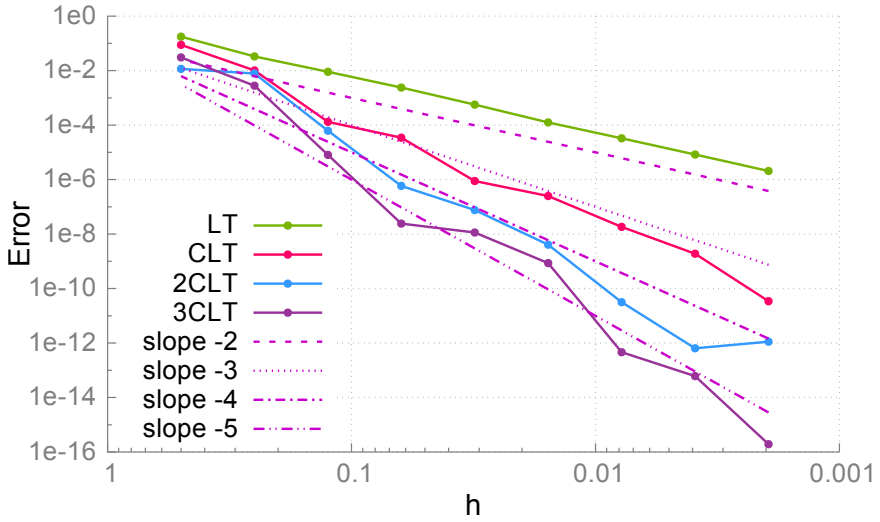


Fig. 6: Error convergence when approximating the function f in (7.1) over a disk using increasing numbers of error correction terms

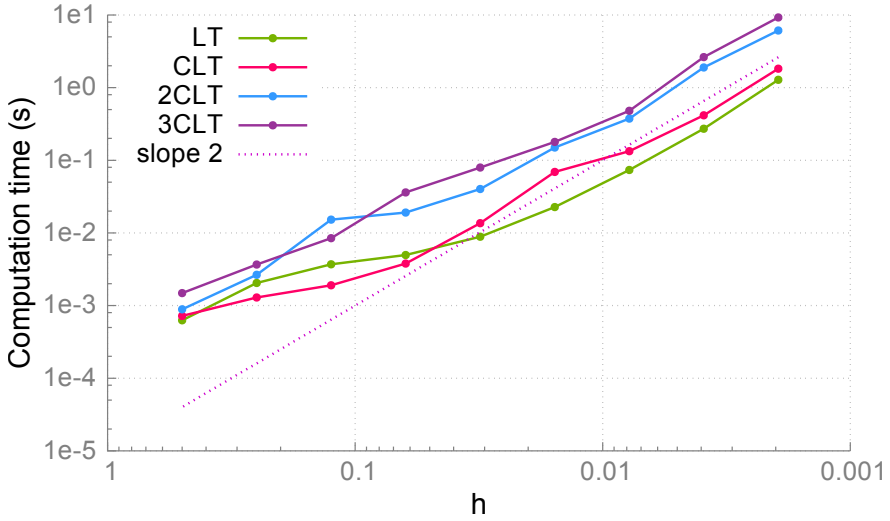


Fig. 7: Computation times when approximating the function f in (7.1) over a disk using increasing numbers of error correction terms

close to the vertex, we plot the ratio of the segment that is inside of the integration region for the trimmed edges of all cells in the segmentation of the disk for $h = \frac{1}{512}$. In Figure 8, we observe that already in the example of the disk, all configurations are present. This shows that the method can handle all trimming situations and the accuracy is not reduced when the trimming curve is close to a vertex in the segmentation.

7.3. Solving the Poisson equation. Finally, we apply our high-order quadrature rule for trimmed surfaces to the solution of a partial differential equation using isogeometric analysis. In

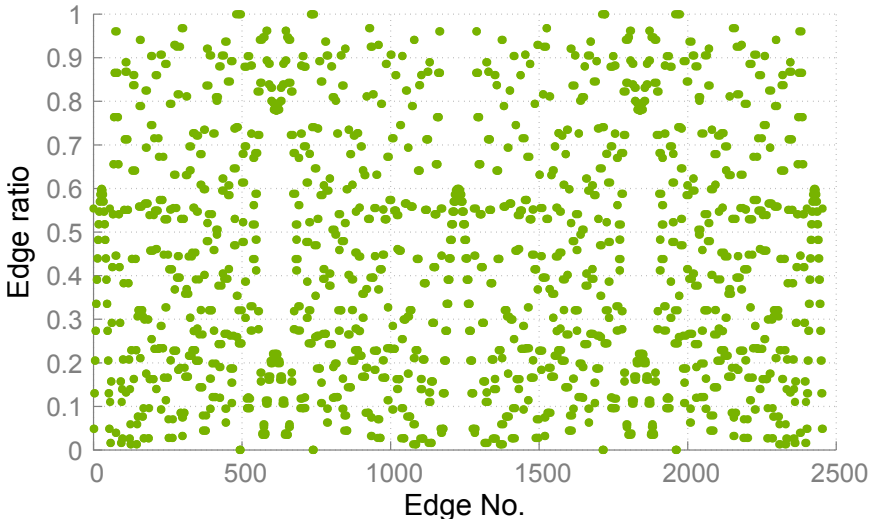


Fig. 8: The edge ratios of the trimmed edges of all cells for $h = \frac{1}{512}$.

particular, we solve the Poisson equation

$$\begin{cases} \Delta u & = f \text{ in } \Omega_\tau, \\ \frac{\partial u}{\partial \boldsymbol{\nu}} & = h \text{ on } \partial\Omega_\tau^{\text{Neumann}}, \\ u & = g \text{ on } \partial\Omega_\tau^{\text{Dirichlet}} \end{cases} . \tag{7.2}$$

As the computation domain Ω_τ, we choose a unit square $B = [0,1]^2$ with a circular hole around its center. The trimming curve is thereby given implicitly by

$$\tau(x,y) = (x-0.5)^2 + (y-0.5)^2 - 0.23^2.$$

We impose homogeneous Neumann boundary conditions on the trimming curve and Dirichlet boundary conditions on the edges of the square. As an exact solution with homogeneous Neumann boundary conditions on the circle we set

$$u(x,y) = \frac{10(x-\frac{1}{2})^2}{(x-\frac{1}{2})^2 + (y-\frac{1}{2})^2} + \tau(x,y)^2(x^{10} + 20y^4 + 100x^6y^7),$$

the function is shown in Figure 9.

We use the standard quadrature rule LT as well as the error-corrected quadrature rules CLT and 2CLT to assemble the system matrix of an isogeometric Galerkin method using B-Splines of different polynomial degrees p. More precisely, we use extended B-Splines [22] in order to achieve numerical stability. This stabilization technique works well for isogeometric methods that use relatively low spline degrees, but it is known to be less efficient for higher degrees [31]. This makes performing experiments with discretizations using high spline degrees challenging. All integrals appearing in the corrected trimmed quadrature rules are approximated using Gaussian quadrature with $p+1$ nodes per direction. The optimal rates of convergence for the employed spline spaces are $p+1$ in the L^2 norm and p in the H^1 seminorm. We therefore consider a quadrature rule to be suitable if it can realize these orders.

Figure 10 reports the L^2-error as well as the H^1-error of the approximate solutions resulting from discretizing with polynomial degrees $p = 2$ through $p = 4$ using our quadrature rules LT, CLT and 2CLT:

- We observe that for $p = 2$ both CLT and 2CLT achieve the optimal cubic order of convergence in the L^2 norm, while LT only achieves second order. All three methods

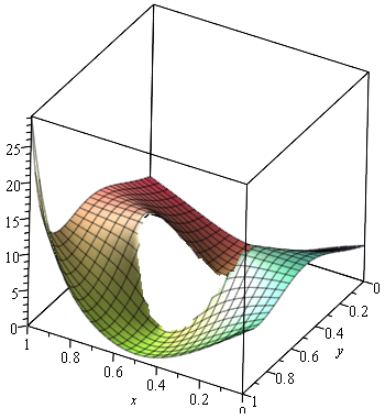


Fig. 9: The exact solution u

achieve the optimal quadratic rate of convergence in the H^1 seminorm. This is to expected, since the overall error is bounded by the quadratic order provided by the LT quadrature rule.

- Comparing the results for degree $p = 3$, we observe that 2CLT is more accurate than CLT and realizes the optimal quartic order of convergence in the L^2 norm, while both methods lead to an optimal cubic order of convergence when measuring the error in the H^1 seminorm.
- Finally, for $p = 4$, the accuracy increases with each correction term and 2CLT is able to realize the optimal quartic rate of convergence in the H^1 seminorm.

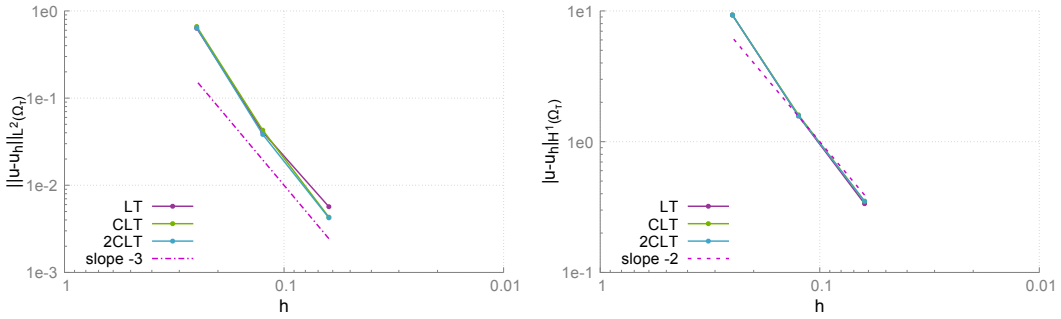
The results indicate that kCLT leads to an optimal rate of convergence in the L^2-norm for spline degrees up to $p = k + 1$. If the error is measured only in the H^1-norm, kCLT is optimal even for spline degrees up to $p = k + 2$.

8. Conclusion and future work. We presented a novel quadrature rule for planar domains that are intersected with an implicitly defined curve. Our method is able to achieve arbitrarily high order and is therefore well suited for high-order geometrically unfitted finite element methods and isogeometric analysis on trimmed CAD models. It as at the same time efficient, accurate and easy to implement and can be integrated into existing software.

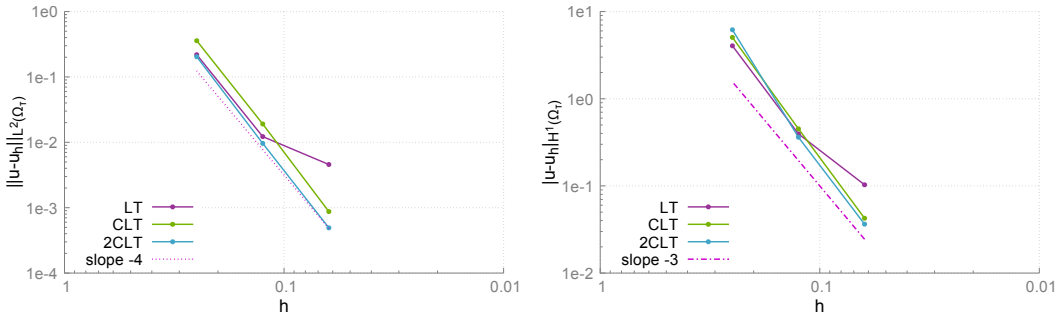
Our method is based on a Taylor expansion of the interpolation between the original integral and an approximation, which makes it necessary to compute derivatives of integrals over moving curves. In order to obtain a quadrature rule of high order, we first proved a transport theorem for moving curves and showed that in the case of implicitly defined moving curves, it can be used to compute any derivative of the integral.

Throughout this paper, we assumed that the function that implicitly defines the boundary of the integration region possesses non-vanishing gradients along its zero-level set, which implies that the boundary is smooth inside the domain. Clearly this is a limitation on the flexibility of the method. In [43, Section 7], we showed how to apply the CLT method with one correction term to a volume intersected with a non-smooth surface by using the observation that the number of singular elements is one order of magnitude smaller than the total number of trimmed elements. A topic for future research is therefore how to extend the high-order quadrature rules to the case of non-smooth boundary curves.

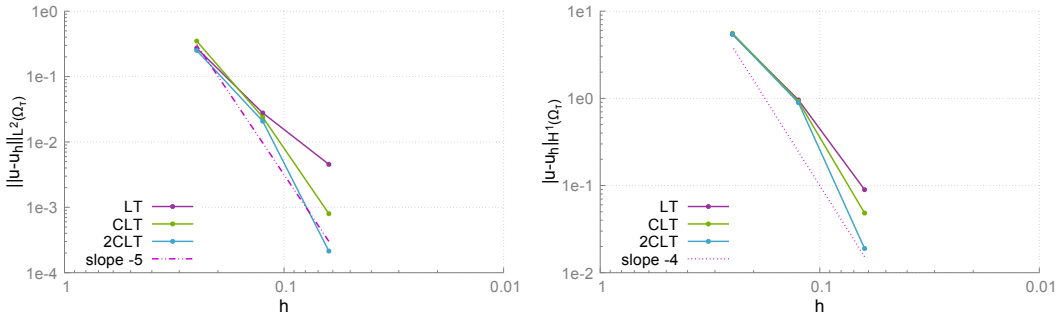
Future work also includes the generalization of the method to the 3D case where the error correction terms include integrals over moving surfaces, moving curves in the integration cells' boundary faces as well as point evaluations at their edges. We provided a guideline on how to apply the techniques presented in this paper to the integration on cut volumes in Remark 5.2. However, additional challenges such as reduced smoothness in the intersection surfaces will need



(a) p=2



(b) p=3



(c) p=4

Fig. 10: Error of the approximate solution to (7.2) in the L^2 norm (left) and the H^1 seminorm (right) using different quadrature rules and polynomial degrees.

to be considered.

Furthermore, in order to arrive at a theoretical guarantee of the convergence order, the remainder term of Taylor expansion needs to be further investigated. For the first order correction term, this analysis has been carried out in [44]. While the details of the proof were quite technical, there seem to be no unsurmountable obstacles for the extension to higher orders.

REFERENCES

[1] P. Antolin, A. Buffa, and M. Martinelli, *Isogeometric analysis on V-reps: First results*, Computer Methods in Applied Mechanics and Engineering, 355 (2019), pp. 976–1002.

[2] S. P. A. Bordas, E. N. Burman, M. G. Larson, and M. A. Olshanskii, eds., *Geometrically Unfitted Finite Element Methods and Applications*, vol. 121 of Lecture Notes in Computational Science and Engineering, Springer, Cham, 2017.

[3] A. Bressan and E. Sande, *Approximation in FEM, DG and IGA: a theoretical comparison*, Numerische Mathematik, 143 (2019), pp. 923–942.
[4] A. Buffa, R. Puppi, and R. Vázquez, *A minimal stabilization procedure for isogeometric methods on trimmed geometries*, arXiv preprint arXiv:1902.04937, (2019).
[5] E. Burman, *Ghost penalty*, Comptes Rendus Mathematique, 348 (2010), pp. 1217 – 1220.
[6] E. Burman, S. Claus, P. Hansbo, M. G. Larson, and A. Massing, *CutFEM: Discretizing geometry and partial differential equations*, International Journal for Numerical Methods in Engineering, 104 (2015), pp. 472–501.
[7] E. Burman and A. Ern, *An unfitted hybrid high-order method for elliptic interface problems*, SIAM Journal on Numerical Analysis, 56 (2018), pp. 1525–1546.
[8] P. Cermelli, E. Fried, and M. E. Gurtin, *Transport relations for surface integrals arising in the formulation of balance laws for evolving fluid interfaces*, Journal of Fluid Mechanics, 544 (2005), p. 339–351.
[9] K. K. Choi and N.-H. Kim, *Structural sensitivity analysis and optimization 1: linear systems*, Springer Science & Business Media, 2006.
[10] L. Coradello, P. Antolin, R. Vázquez, and A. Buffa, *Adaptive isogeometric analysis on two-dimensional trimmed domains based on a hierarchical approach*, Computer Methods in Applied Mechanics and Engineering, 364 (2020), p. 112925.
[11] T. Cui, W. Leng, H. Liu, L. Zhang, and W. Zheng, *High-order numerical quadratures in a tetrahedron with an implicitly defined curved interface*, ACM Transactions on Mathematical Software (TOMS), 46 (2020), pp. 1–18.
[12] F. de Prenter, C. Verhoosel, and E. van Brummelen, *Preconditioning immersed isogeometric finite element methods with application to flow problems*, Computer Methods in Applied Mechanics and Engineering, 348 (2019), pp. 604–631.
[13] F. de Prenter, C. V. Verhoosel, E. van Brummelen, J. Evans, C. Messe, J. Benzaken, and K. Maute, *Multigrid solvers for immersed finite element methods and immersed isogeometric analysis*, Computational Mechanics, 65 (2020), pp. 807–838.
[14] K. Deckelnick, C. M. Elliott, and T. Ranner, *Unfitted finite element methods using bulk meshes for surface partial differential equations*, SIAM Journal on Numerical Analysis, 52 (2014), pp. 2137–2162.
[15] S. C. Divi, C. V. Verhoosel, F. Auricchio, A. Reali, and E. H. van Brummelen, *Error-estimate-based adaptive integration for immersed isogeometric analysis*, Computers & Mathematics with Applications, (2020).
[16] J. A. Evans, Y. Bazilevs, I. Babuška, and T. J. Hughes, *n-widths, sup–infs, and optimality ratios for the k-version of the isogeometric finite element method*, Computer Methods in Applied Mechanics and Engineering, 198 (2009), pp. 1726 – 1741.
[17] C. Giannelli, T. Kanduč, F. Pelosi, and H. Speleers, *An immersed-isogeometric model: Application to linear elasticity and implementation with THBox-splines*, Journal of Computational and Applied Mathematics, 349 (2019), pp. 410–423.
[18] R. Glowinski, T.-W. Pan, and J. Periaux, *A fictitious domain method for dirichlet problem and applications*, Computer Methods in Applied Mechanics and Engineering, 111 (1994), pp. 283 – 303.
[19] R. Goldman, *Curvature formulas for implicit curves and surfaces*, Computer Aided Geometric Design, 22 (2005), pp. 632 – 658.
[20] J. Hakenberg and U. Reif, *On the volume of sets bounded by refinable functions*, Applied Mathematics and Computation, 272 (2016), pp. 2 – 19. Subdivision, Geometric and Algebraic Methods, Isogeometric Analysis and Refinability.
[21] T. Hughes, J. Cottrell, and Y. Bazilevs, *Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement*, Computer Methods in Applied Mechanics and Engineering, 194 (2005), pp. 4135–4195.
[22] K. Höllig, U. Reif, and J. Wipper, *Weighted extended B-spline approximation of Dirichlet problems*, SIAM Journal on Numerical Analysis, 39 (2001), pp. 442–462.
[23] C.-K. Im and S.-K. Youn, *The generation of 3D trimmed elements for NURBS-based isogeometric analysis*, International Journal of Computational Methods, 15 (2018), pp. 1–33.
[24] M. Joulaian, S. Hubrich, and A. Düster, *Numerical integration of discontinuities on arbitrary domains based on moment fitting*, Computational Mechanics, 57 (2016), pp. 979–999.
[25] H. Kim, Y. Seo, and S. . Youn, *Isogeometric analysis for trimmed CAD surfaces*, Computer Methods in Applied Mechanics and Engineering, 198 (2009), pp. 2982–2995.
[26] L. Kudela, N. Zander, S. Kollmannsberger, and E. Rank, *Smart octrees: Accurately integrating discontinuous functions in 3D*, Computer Methods in Applied Mechanics and Engineering, 306 (2016), pp. 406 – 426.
[27] L. Leidinger, M. Breitenberger, A. Bauer, S. Hartmann, R. Wüchner, K.-U. Bletzinger, F. Duddeck, and L. Song, *Explicit dynamic isogeometric B-rep analysis of penalty-coupled trimmed NURBS shells*, Computer Methods in Applied Mechanics and Engineering, 351 (2019), pp. 891–927.
[28] T. Lin, Y. Lin, and X. Zhang, *Partially penalized immersed finite element methods for elliptic interface problems*, SIAM Journal on Numerical Analysis, 53 (2015), pp. 1121–1144.
[29] T. Ludescher, S. Gross, and A. Reusken, *A multigrid method for unfitted finite element discretizations of elliptic interface problems*, SIAM Journal on Scientific Computing, 42 (2020), pp. A318–A342.
[30] A. Mantzaflaris, F. Scholz, and others (see website), *G+Smo (Geometry plus Simulation modules) v0.8.1*. http://gs.jku.at/gismo, 2020.

[31] B. Marussig, R. Hiemstra, and T. Hughes, *Improved conditioning of isogeometric analysis matrices for trimmed geometries*, Computer Methods in Applied Mechanics and Engineering, 334 (2018), pp. 79–110.
[32] B. Marussig and T. Hughes, *A review of trimming in isogeometric analysis: Challenges, data exchange and simulation aspects*, Archives of Computational Methods in Engineering, 25 (2017), pp. 1–69.
[33] F. Massarwi, P. Antolin, and G. Elber, *Volumetric untrimming: Precise decomposition of trimmed trivariates into tensor products*, Computer Aided Geometric Design, 71 (2019), pp. 1–15.
[34] R. Massjung, *An unfitted discontinuous Galerkin method applied to elliptic interface problems*, SIAM Journal on Numerical Analysis, 50 (2012), pp. 3134–3162.
[35] S. Mousavi and N. Sukumar, *Numerical integration of polynomials and discontinuous functions on irregular convex polygons and polyhedrons*, Computational Mechanics, 47 (2011), pp. 535–554.
[36] A. Nagy and D. Benson, *On the numerical integration of trimmed isogeometric elements*, Computer Methods in Applied Mechanics and Engineering, 284 (2015), pp. 165–185.
[37] T. S. Newman and H. Yi, *A survey of the marching cubes algorithm*, Computers & Graphics, 30 (2006), pp. 854 – 879.
[38] M. A. Olshanskii and D. Safin, *Numerical integration over implicitly defined domains for higher order unfitted finite element methods*, Lobachevskii Journal of Mathematics, 37 (2016), pp. 582–596.
[39] N. Patrikalakis and T. Maekawa, *Shape Interrogation for Computer Aided Design and Manufacturing*, Springer, Berlin Heidelberg, 2002.
[40] M. Ruess, D. Schillinger, A. Özcan, and E. Rank, *Weak coupling for isogeometric analysis of non-matching and trimmed multi-patch geometries*, Computer Methods in Applied Mechanics and Engineering, 269 (2014), pp. 46–71.
[41] R. I. Saye, *High-order quadrature methods for implicitly defined surfaces and volumes in hyperrectangles*, SIAM Journal on Scientific Computing, 37 (2015), pp. A993–A1019.
[42] D. Schillinger and M. Ruess, *The finite cell method: A review in the context of higher-order structural analysis of CAD and image-based geometric models*, Archives of Computational Methods in Engineering, 22 (2015), pp. 391–455.
[43] F. Scholz and B. Jüttler, *Numerical integration on trimmed three-dimensional domains with implicitly defined trimming surfaces*, Computer Methods in Applied Mechanics and Engineering, 357 (2019), p. 112577.
[44] F. Scholz, A. Mantzaflaris, and B. Jüttler, *First order error correction for trimmed quadrature in isogeometric analysis*, in Advanced Finite Element Methods with Applications, vol. 128 of Lecture Notes in Computational Science and Engineering, Cham, 2019, Springer, pp. 297–321.
[45] Y. Sudhakar and W. A. Wall, *Quadrature schemes for arbitrary convex/concave volumes and integration of weak form in enriched partition of unity methods*, Computer Methods in Applied Mechanics and Engineering, 258 (2013), pp. 39 – 54.
[46] V. Thiagarajan and V. Shapiro, *Adaptively weighted numerical integration over arbitrary domains*, Computers & Mathematics with Applications, 67 (2014), pp. 1682 – 1702.
[47] V. Thiagarajan and V. Shapiro, *Adaptively weighted numerical integration in the finite cell method*, Computer Methods in Applied Mechanics and Engineering, 311 (2016), pp. 250–279.
[48] C. Wang, X. Zhu, and X. Zhang, *Quasi-conforming analysis method for trimmed CAD surfaces*, European Journal of Mechanics-A/Solids, 81 (2020), p. 103959.
[49] T. Yang, A. Qarariyah, H. Kang, and J. Deng, *Numerical integration over implicitly defined domains with topological guarantee*, Communications in Mathematics and Statistics, 7 (2019), pp. 459–474.
[50] L. Zhang, A. Gerstenberger, X. Wang, and W. K. Liu, *Immersed finite element method*, Computer Methods in Applied Mechanics and Engineering, 193 (2004), pp. 2051 – 2067. Flow Simulation and Modeling.
[51] X. Zhu, Z. Ma, and P. Hu, *Nonconforming isogeometric analysis for trimmed CAD geometries using finite-element tearing and interconnecting algorithm*, Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, 231 (2017), pp. 1371–1389.