An algebraic cubature formula on curvilinear polygons *

G. Santin, A. Sommariva[1] and M. Vianello[1]

May 3, 2011

Abstract

We implement in Matlab a Gauss-like cubature formula on bivariate domains whose boundary is a piecewise smooth Jordan curve (curvilinear polygons). The key tools are Green's integral formula, together with the recent software package `Chebfun` to approximate the boundary curve close to machine precision by piecewise Chebyshev interpolation. Several tests are presented, including some comparisons of this new routine `ChebfunGauss` with the recent `SplineGauss` that approximates the boundary by splines.

Keywords: algebraic cubature, Chebfun, Green's formula.

1 Introduction.

The problem of integrating numerically at high precision a function over a general bidimensional domain whose boundary is a piecewise smooth Jordan curve (curvilinear polygon), is not a trivial task. General purpose cubature packages typically require that the domain be splitted by the user into simpler geometric elements, and are written in Fortran or C (cf., e.g., the Cubpack package [6]). Even worse is the situation with standard integrators like Matlab's `dblquad`, when applied trivially by multiplying with the characteristic function of the domain, since in such a case an artificial discontinuity at the boundary is introduced, which causes inefficiency and often unreliability at high precision. A substantial improvement has been given by the recent Matlab program `TwoD` by Shampine, which is much more reliable and efficient, but still requires that the domain be splitted into generalized rectangles or sectors; cf. [13].

In some recent papers, a cubature formula over polygons, and more generally bivariate compact domains whose boundary can be well approximated parametrically by splines, has been proposed (see [14, 15]). Such a formula

$$\sum_{\lambda} w_{\lambda} f(x_{\lambda}, y_{\lambda}) \approx I_{\Omega}(f) = \iint_{\Omega} f(x,y)\, dx\, dy\ ,\ \ \Omega \subset \mathbb{R}^2 \tag{1}$$

*Work supported by the research project "Interpolation and Extrapolation: new algorithms and applications" of the University of Padova and by the GNCS-INdAM.

[1]Dept. of Pure and Applied Mathematics, University of Padova, Italy
e-mail: alvise, marcov@math.unipd.it

is in particular exact in $\mathbb{P}^2_{2n-1}$ (the space of bivariate polynomials of degree at most $2n-1$), stable (since $\sum_\lambda |w_\lambda|$ is bounded with n), the nodes and weights being explicitly known in terms of the univariate Gauss-Legendre ones.

The key point is the Green's integral formula (see, e.g., [1])

$$I_\Omega(f) = \oint_{\partial\Omega} \mathcal{F}_f(x,y)\,dy\,,\ \mathcal{F}_f(x,y) = \int f(x,y)\,dx\,,\ f \in C(\Omega) \qquad (2)$$

that suggests to approximate the x-primitive $\mathcal{F}_f$ via a suitable Gauss-Legendre discretization and then integrate along the boundary, again by a Gauss-Legendre rule.

In this paper we apply the same idea, using the recent software package `Chebfun` by Trefethen and others [18], which works in the Matlab environment, to approximate the boundary curve close to machine precision by piecewise Chebyshev interpolation. What we obtain is intermediate between a cubature algorithm and a cubature formula: indeed, the final result is an algebraic cubature formula with a fixed degree of exactness, but a key step is the adaptive approximation of the domain boundary. We compare the performance of our formula with the standard Matlab integrator `dblquad` and the routine `SplineGauss` [16] on several domains, showing that the new routine `ChebfunGauss` can achieve a given accuracy with a much lower number of function evaluations. Moreover, it turns out to be competitive also with the adaptive vectorized cubature code `TwoD` by Shampine [13].

2 Piecewise polynomial boundaries

In this section we briefly describe how to construct an algebraic cubature formula using the `Chebfun` system. The theorem that follows adapts a result obtained in [15] (cf. [12]). If the boundary of the compact domain $\Omega \subset \mathbb{R}^2$ is defined parametrically by two continuous functions $x, y : [a,b] \to \mathbb{R}$ (such that $x(a) = x(b)$, $y(a) = y(b)$), then `Chebfun` provides two (globally continuous) piecewise polynomial interpolants $\tilde{x}, \tilde{y}$ such that the relative errors in infinite norm between x and $\tilde{x}$, y and $\tilde{y}$, are close to the machine precision `eps`.

Consequently, first we will construct a cubature rule based on the approximation of the boundary by a piecewise polynomial parametric curve, and then analyze the cubature error w.r.t. the integral on the original domain. The first step is made with the following

Theorem 1 *Let $K \subset \mathbb{R}^2$ be a compact domain (the closure of a bounded and simply connected open set), whose boundary ∂K is a Jordan piecewise polynomial parametric curve, $S(t)$, $t \in [t_1, t_L]$, given counterclockwise by a sequence of polynomial parametric curves $S_i = (S_{i,1}, S_{i,2})$, with $S_{i,1} \in \mathbb{P}_{i_1}$, $S_{i,2} \in \mathbb{P}_{i_2}$ defined in the interval $[t_i, t_{i+1}]$, and "breakpoints"*

$$V_i = S_i(t_i) = (S_{i,1}(t_i), S_{i,2}(t_i))\ ,\quad i = 1, \ldots, L \qquad (3)$$

with $S_i(t_{i+1}) = S_{i+1}(t_{i+1})$, $i = 1, \ldots, L-1$, $S_L(t_{L+1}) = S_1(t_1)$ (i.e., $V_{L+1} = V_1$). Furthermore, let $\mathcal{R} = [x_1, x_2] \times [y_1, y_2]$ be the minimal rectangle containing K, $f \in C(\mathcal{R})$, $\xi \in [x_1, x_2]$, $\{\tau^s_k\}_{k=1,\ldots,s}$ and $\{\omega^s_k\}_{k=1,\ldots,s}$ the nodes and weights of the Gauss-Legendre rule in $[-1,1]$ (having degree of exactness $2s-1$).

The cubature rule

$$I_{2n-1}(f;S) = \sum_{i=1}^{L} \sum_{j=1}^{n} \sum_{k=1}^{n_i} w_{ijk} f(x_{ijk}, y_{ik}) \tag{4}$$

with

$$x_{ijk} = \frac{S_{i,1}(q_i(\tau_k^{n_i})) - \xi}{2} \tau_j^n + \frac{S_{i,1}(q_i(\tau_k^{n_i})) + \xi}{2}, \; y_{ik} = S_{i,2}(q_i(\tau_k^{n_i})) \tag{5}$$

$$w_{ijk} = \omega_j^n \, \omega_k^{n_i} \left(\frac{S_{i,1}(q_i(\tau_k^{n_i})) - \xi}{2} \right) S'_{i,2}(q_i(\tau_k^{n_i})) \, \frac{\Delta t_i}{2} \tag{6}$$

where, denoting by $\lceil \cdot \rceil$ *the ceiling operator,*

$$q_i(s) = \frac{\Delta t_i}{2} s + \frac{t_i + t_{i+1}}{2}, \; \Delta t_i = t_{i+1} - t_i \tag{7}$$

$$n_i = \left\lceil \frac{(2n-1)\max(i_1, i_2) + i_1 + i_2}{2} \right\rceil \; , \;\; i_\nu = deg(S_{i,\nu}) \; , \; \nu = 1, 2 \tag{8}$$

is exact on K *for every bivariate polynomial* $f \in \mathbb{P}^2_{2n-1}$.

Proof. Denoting by $V_i \frown V_{i+1}$ the part of the curve between the two breakpoints V_i and V_{i+1}, by Green's theorem and $\partial K = \cup_{i=1}^{L} V_i \frown V_{i+1}$ we have

$$I_K(f) = \int_K f(x,y)\, dx\, dy = \oint_{\partial K} \mathcal{F}_f(x,y)\, dy = \sum_{i=1}^{L} \int_{V_i \frown V_{i+1}} \mathcal{F}_f(x,y)\, dy \tag{9}$$

where $\mathcal{F}_f$ is an x-primitive of f i.e., for a fixed $\xi \in [x_1, x_2]$

$$\mathcal{F}_f(x,y) = \int_{\xi}^{x} f(s,y)\, ds \tag{10}$$

In view of the parametrization of the boundary, from (10)

$$\begin{aligned} \sum_{i=1}^{L} \int_{V_i \frown V_{i+1}} \mathcal{F}_f(x,y)\, dy &= \sum_{i=1}^{L} \int_{t_i}^{t_{i+1}} \mathcal{F}_f(S_{i,1}(t), S_{i,2}(t))\, S'_{i,2}(t)\, dt \\ &= \sum_{i=1}^{L} \int_{t_i}^{t_{i+1}} \left(\int_{\xi}^{S_{i,1}(t)} f(\eta, S_{i,2}(t))\, d\eta \right) S'_{i,2}(t)\, dt \end{aligned} \tag{11}$$

By operating for each univariate integral an affine change of variables, $\eta = \eta(\tau; t) = (S_{i,1}(t) - \xi)\tau/2 + (S_{i,1}(t) + \xi)/2$ and $t = q_i(u)$, cf. (7), we have

$$\begin{aligned} I_K(f) &= \sum_{i=1}^{L} \int_{-1}^{1} \int_{-1}^{1} f\left(\frac{S_{i,1}(q_i(u)) - \xi}{2} \tau + \frac{S_{i,1}(q_i(u)) + \xi}{2}, S_{i,2}(q_i(u)) \right) \\ &\cdot \left(\frac{S_{i,1}(q_i(u)) - \xi}{2} \right) S'_{i,2}(q_i(u))\, \frac{\Delta t_i}{2}\, d\tau\, du \end{aligned} \tag{12}$$

Now, if $f \in \mathbb{P}^2_{2n-1}$, it is easily checked that the integrand on the r.h.s. of (12) is a bivariate polynomial having in the variable τ at most degree $2n-1$ and that, w.r.t. the variable u, is a polynomial of degree

$$(2n-1)\max\left(\deg(S_{i,1}),\deg(S_{i,2})\right)+\deg(S_{i,1})+\deg(S_{i,2})-1$$

As consequence the integral can be computed exactly by a tensorial Gauss-Legendre rule having $n \times n_i$ nodes where n_i is the smallest integer such that

$$2n_i-1 \geq (2n-1)\max\left(\deg(S_{i,1}),\deg(S_{i,2})\right)+\deg(S_{i,1})+\deg(S_{i,2})-1$$

i.e.,

$$n_i = \left\lceil \frac{(2n-1)\max\left(\deg(S_{i,1}),\deg(S_{i,2})\right)+\deg(S_{i,1})+\deg(S_{i,2})}{2} \right\rceil$$

From (12) it is straightforward to determine the nodes and the weights of the cubature rule. If $\{\tau^s_k\}_{k=1,\dots,s}$, $\{\omega^s_k\}_{k=1,\dots,s}$ are, respectively, the nodes and weights of the Gauss-Legendre rule in $[-1,1]$ (having degree of exactness $2s-1$), then

$$\begin{aligned} I_K(f) &= \sum_{i=1}^{L}\sum_{j=1}^{n}\sum_{k=1}^{n_i}\omega^n_j\,\omega^{n_i}_k\, f\left(\frac{S_{i,1}(q_i(\tau^{n_i}_k))-\xi}{2}\tau^n_j+\frac{S_{i,1}(q_i(\tau^{n_i}_k))+\xi}{2}, S_{i,2}(q_i(\tau^{n_i}_k))\right) \\ &\cdot \left(\frac{S_{i,1}(q_i(\tau^{n_i}_k))-\xi}{2}\right) S'_{i,2}(q_i(\tau^{n_i}_k))\,\frac{\Delta t_i}{2} \end{aligned} \quad (13)$$

that implies

$$\begin{aligned} x_{ijk} &= \frac{S_{i,1}(q_i(\tau^{n_i}_k))-\xi}{2}\tau^n_j+\frac{S_{i,1}(q_i(\tau^{n_i}_k))+\xi}{2} \\ y_{ik} &= S_{i,2}(q_i(\tau^{n_i}_k)) \\ w_{ijk} &= \omega^n_j\,\omega^{n_i}_k\left(\frac{S_{i,1}(q_i(\tau^{n_i}_k))-\xi}{2}\right) S'_{i,2}(q_i(\tau^{n_i}_k))\,\frac{\Delta t_i}{2} \end{aligned} \quad (14)$$

with $i=1,\dots,L$, $j=1,\dots,n$, $k=1,\dots,n_i$. $\square$

Remark 1 In general the nodes are not contained in the cubature domain Ω but only in the *minimal* rectangle $\mathcal{R} \supseteq K$ with sides parallel to the axes. This fact comes directly from the definition of the cubature nodes. From (5) and (7), being $\tau^{n_i}_k \in [-1,1]$, we have $q_i(s) \in [t_i, t_{i+1}]$. Furthermore, it is easily seen that x_{ijk} belongs to the minimal interval containing $S_{i,1}(q_i(\tau^{n_i}_k))$ and ξ, implying that (x_{ijk},y_{ik}) is in the segment connecting $(\xi, S_{i,2}(q_i(\tau_k{}^{n_i})))$ to the boundary point $(S_{i,1}(q_i(\tau_k{}^{n_i})), S_{i,2}(q_i(\tau_k{}^{n_i})))$. Since the latter are both in the rectangle $\mathcal{R}$ necessarily every (x_{ijk},y_{ik}) is in $\mathcal{R}$ and of course this the reason why we require that the integrand f is defined in $\mathcal{R}$.

However, as described in [14], such assumption is not necessary in domains in which there exists a straightline l such that

Property N (normal domain w.r.t. a line)

(A1) $l \cap K$ is connected;

(A2) every segment q orthogonal to l is such that $q \cap K$ is connected.

It is not difficult to show that if K is a convex set, taking as *base-line* l the straightline connecting two points P_1, $P_2 \in \partial K$ such that the segment $[P_1, P_2]$ is a diameter, the assumptions (A1) and (A2) are verified.

When Property N holds, a change of coordinates (rotation) so that l becomes parallel to the new y-axis implies that all the cubature nodes (x_{ijk}, y_{ik}) are in K, and that all the weights w_{ijk} are nonnegative, taking as ξ in Theorem 1 the intersection point of l with the new x-axis.

Indeed, when the boundary point $(S_{i,1}(q_i(\tau_k{}^{n_i})), S_{i,2}(q_i(\tau_k{}^{n_i})))$ is on the right (resp. left) of the base-line, i.e., $S_{i,1}(q_i(\tau_k{}^{n_i})) \geq \xi$ (resp. $S_{i,1}(q_i(\tau_k{}^{n_i})) \leq \xi$), then the tangent vector to the curve remains in the first and second (resp. third and fourth) quadrant, and thus $S'_{i,2}(q_i(\tau_k{}^{n_i}))$ is nonnegative (resp. nonpositive); see [14, 15] for more details.

To have an idea of the role of the base-line in the distribution of the cubature nodes, we suggest to have a look at the figures in the last section.

Remark 2 From the previous theorem, the cubature formula having degree of exactness $2n-1$ over a domain K defined parametrically by L polynomial curves $(S_{i,1}, S_{i,2})$, has $n\sum_{i=1}^{L} n_i$ nodes where

$$n_i = \left\lceil \frac{(2n-1)\max\left(\deg(S_{i,1}), \deg(S_{i,2})\right) + \deg(S_{i,1}) + \deg(S_{i,2})}{2} \right\rceil$$

Consequently, if $\sigma = \max_{i=1,\ldots,L}\left(\max\left(\deg(S_{i,1}), \deg(S_{i,2})\right)\right)$ then

$$n+1 = \left\lceil \frac{2n+1}{2} \right\rceil \leq n_i \leq \left\lceil \frac{(2n+1)\sigma}{2} \right\rceil \leq (n+1)\sigma$$

and

$$n(n+1)L \leq n\sum_{i=1}^{L} n_i \leq n(n+1)L\sigma$$

It is worth noticing that if the user needs a cubature rule having *even* degree of exactness M, the cubature rule will actually have degree of excatness $M+1$. This is due to the fact that the Gauss-Legendre formula has odd degree of exactness $2n-1$.

2.1 Stability and error estimates

Let Ω be a compact domain whose boundary is a Jordan (simple and closed) curve defined parametrically by two piecewise smooth functions $x(t)$, $y(t)$ that are not piecewise polynomials

$$\partial\Omega = \{P(t) = (x(t), y(t))\,,\ t \in [a, b]\}\ ,\ P(a) = P(b) \tag{15}$$

The `Chebfun` system provides two piecewise polynomials (piecewise interpolating at Chebyshev-Lobatto nodes) $\tilde{x}$, $\tilde{y} : [a, b] \to \mathbb{R}$ such that

$$\|x - \tilde{x}\|_\infty \leq \varepsilon\|x\|_\infty,\ \|y - \tilde{y}\|_\infty \leq \varepsilon\|y\|_\infty \tag{16}$$

ε being by default the machine precision. Denoting by $\tilde{\Omega}$ is the domain whose boundary is defined as

$$\partial\tilde{\Omega} = \{\tilde{P}(t) = (\tilde{x}(t), \tilde{y}(t))\,,\ t \in [a,b]\} \tag{17}$$

it is natural to approximate $I_\Omega(f)$ with $I_{\tilde{\Omega}}(f)$. Observe that by interpolation $\partial\tilde{\Omega}$ is a closed curve, $\tilde{P}(a) = \tilde{P}(b)$; we shall discuss below conditions ensuring that such a piecewise polynomial curve is still a Jordan curve. On the other hand, stability issues of cubature rules require the boundedness of the sum of the weights absolute values, independently of the degree of exactness. Here we will first give such an estimate and then use it for giving an upper bound to $|I_\Omega(f) - I_{2n-1}(f;\tilde{P})|$.

Theorem 2 *Under the assumptions of Theorem 1, denoting by $\ell(\partial K)$ the length of the boundary of the domain K, we have*

$$\sum_{i=1}^{L}\sum_{j=1}^{n}\sum_{k=1}^{n_i} |w_{ijk}| \le (x_2 - x_1)\,\ell_n \ \textit{with} \ \lim_n \ell_n = \ell(\partial K)$$

Proof. With the notation used in Theorem 1, since (x_{ijk}, y_{ik}) are points of the minimal rectangle $\mathcal{R} = [x_1, x_2] \times [y_1, y_2]$ containing K (see Remark 1) and $\xi \in [x_1, x_2]$, we have $|x_{ijk} - \xi| \le x_2 - x_1$. Furthermore, since the Gauss-Legendre weights are positive and the sum of their absolute values is 2, we have from (6)-(8)

$$\begin{aligned}
\sum_{i=1}^{L}\sum_{j=1}^{n}\sum_{k=1}^{n_i} |w_{ijk}| &= \sum_{i=1}^{L}\sum_{j=1}^{n}\sum_{k=1}^{n_i} \omega_j^n\, \omega_k^{n_i} \frac{|S_{i,1}(q_i(\tau_k^{n_i})) - \xi|}{2} |S'_{i,2}(q_i(\tau_k^{n_i}))| \frac{\Delta t_i}{2} \\
&\le 2\,\frac{(x_2 - x_1)}{2} \sum_{i=1}^{L}\sum_{k=1}^{n_i} \omega_k^{n_i} |S'_{i,2}(q_i(\tau_k^{n_i}))| \frac{\Delta t_i}{2} \\
&\le (x_2 - x_1) \sum_{i=1}^{L}\sum_{k=1}^{n_i} \omega_k^{n_i} \sqrt{(S'_{i,1}(q_i(\tau_k^{n_i})))^2 + (S'_{i,2}(q_i(\tau_k^{n_i})))^2}\, \frac{\Delta t_i}{2}
\end{aligned} \tag{18}$$

Now, the quantity

$$\ell_{i,n} = \sum_{k=1}^{n_i} \omega_k^{n_i} \sqrt{(S'_{i,1}(q_i(\tau_k^{n_i})))^2 + (S'_{i,2}(q_i(\tau_k^{n_i})))^2}\, \frac{\Delta t_i}{2}$$

corresponds to compute the length of arc $V_i \frown V_{i+1}$ by the Gauss-Legendre rule. Observe that $\ell_{i,n}$ depends on n through n_i, cf. (8). Setting $\ell_n = \sum_{i=1}^{L} \ell_{i,n}$ we have

$$\sum_{i=1}^{L}\sum_{j=1}^{n}\sum_{k=1}^{n_i} |w_{ijk}| \le (x_2 - x_1) \sum_{i=1}^{L} \ell_{i,n} = (x_2 - x_1)\,\ell_n$$

The fact that $\lim_n \ell_n = \ell(\partial K)$ is due to the convergence properties of Gauss-Legendre quadrature. □

We can now bound the error made in approximating $I_\Omega(f)$ with $I_{2n-1}(f;\tilde{P})$. Setting $\|f\|_J = \max_{x\in J}|f(x)|$, $E_m(f;J) = \min_{p\in\mathbb{P}^2_m}\|f-p\|_J$ for any compact set J and continuous function f, $\mu(A) = \mathrm{meas}(A)$ and $A\Delta B = (A\cup B)\setminus(A\cap B)$ for any couple of measurable sets, we have

Theorem 3 *Let Ω and $\tilde{\Omega}$ be as in (15)-(17). Under the assumptions of Theorem 1 for $K=\tilde{\Omega}$, if $f\in C(\Omega\cup\tilde{\Omega})$ the following cubature error estimate holds*

$$|I_\Omega(f) - I_{2n-1}(f;\tilde{P})| \leq (\mu(\tilde{\Omega}) + (x_2-x_1)\ell_n)E_{2n-1}(f;\mathcal{R}) + \|f\|_{\Omega\Delta\tilde{\Omega}}\,\mu(\Omega\Delta\tilde{\Omega})$$

where $\lim \ell_n = \ell(\partial\tilde{\Omega})$ *(cf. Theorem 2).*

Proof. By the triangle inequality,

$$|I_\Omega(f) - I_{2n-1}(f;\tilde{P})| \leq |I_\Omega(f) - I_{\tilde{\Omega}}(f)| + |I_{\tilde{\Omega}}(f) - I_{2n-1}(f;\tilde{P})|$$

First, we observe that

$$|I_\Omega(f) - I_{\tilde{\Omega}}(f)| \leq \|f\|_{\Omega\Delta\tilde{\Omega}}\,\mu(\Omega\Delta\tilde{\Omega})$$

Next, if $p^*_{2n-1}\in\mathbb{P}^2_{2n-1}$ is such that $E_{2n-1}(f;\mathcal{R}) = \|f-p^*_{2n-1}\|_{\mathcal{R}}$ then, since the cubature formula has degree of exactness $2n-1$ on $\tilde{\Omega}$

$$\begin{aligned}
|I_{\tilde{\Omega}}(f) - I_{2n-1}(f;\tilde{P})| &\leq |I_{\tilde{\Omega}}(f) - I_{\tilde{\Omega}}(p^*_{2n-1})| + |I_{\tilde{\Omega}}(p^*_{2n-1}) - I_{2n-1}(p^*_{2n-1};\tilde{P})| \\
&+ |I_{2n-1}(p^*_{2n-1};\tilde{P}) - I_{2n-1}(f;\tilde{P})| \\
&\leq \left(\mu(\tilde{\Omega}) + \sum_{i=1}^{L}\sum_{j=1}^{n}\sum_{k=1}^{n_i}|w_{ijk}|\right)E_{2n-1}(f;\mathcal{R}) \qquad (19)
\end{aligned}$$

that easily gives the error estimate by Theorem 2. $\square$

Remark 3 In view of (16), we have that

$$\Omega\cup\tilde{\Omega}\subseteq\Omega + B[0,r(\varepsilon)]\ ,\ \ \Omega\Delta\tilde{\Omega}\subseteq\partial\Omega + B[0,r(\varepsilon)]$$

where

$$r(\varepsilon) = \varepsilon\sqrt{\|x\|^2_\infty + \|y\|^2_\infty} \qquad (20)$$

and $B[0,r]$ denotes the closed disk centered at the origin with radius r. From these inclusions easily follows that $\mu(\Omega\Delta\tilde{\Omega}) = \mathcal{O}(\varepsilon)$, and $\mu(\tilde{\Omega}) = \mu(\Omega) + \mathcal{O}(\varepsilon)$ since $\tilde{\Omega}\setminus(\Omega\Delta\tilde{\Omega})\subseteq\Omega\subseteq\tilde{\Omega}\cup(\Omega\Delta\tilde{\Omega}) = \Omega\cup\tilde{\Omega}$. Moreover, when Ω satisfies Property N above, after the change of variables (rotation) all the cubature nodes fall in $\Omega + B[0,r(\varepsilon)]$, so that the latter can substitute the rectangle $\mathcal{R}$ in Theorem 3.

The problem of bounding ℓ_n in terms of the original domain Ω is more delicate. Indeed, by Theorem 2 we have that $\lim_n \ell_n = \ell(\partial\tilde{\Omega})$, but to know that $\ell(\partial\tilde{\Omega})\approx\ell(\partial\Omega)$ we should ensure that not only the curve $\partial\Omega$, but also its tangent vectors are piecewise approximated, at least in the L^1 norm. The theory of Chebyshev interpolation, on which the `Chebfun` package is based, tells us that we have also L^1-convergence to the derivatives if x', y' are at least piecewise Hölder continuous. This comes from the chain of estimates

$$\|u' - (I_N u)'\|_{L^1(\alpha,\beta)} \leq (b-a)\|1/w\|_\infty\|u' - (I_N u)'\|_{L^2_w(\alpha,\beta)} = \mathcal{O}\left(N^{1-m}\right) \quad (21)$$

valid for $u \in C^m[\alpha, \beta]$, $m > 1$, where $[\alpha, \beta] \subseteq [a, b]$ is any subinterval of smoothness, $I_N u$ is the interpolant of degree N at the Chebyshev-Lobatto nodes, and w is the Chebyshev weight function for $[\alpha, \beta]$; cf., e.g., [5]. In such a case we can finally write an error estimate like

$$|I_\Omega(f) - I_{2n-1}(f; \tilde{P})| = \mathcal{O}\left(E_{2n-1}(f; J) + \varepsilon\right)$$

with $J = \mathcal{R}$, or even $J = \Omega + B[0, r(\varepsilon)]$ when Ω satisfies Property N.

In order to apply Theorem 3, we should ensure that the approximate boundary $\partial\tilde{\Omega}$ is a Jordan curve, namely that it is a *simple* curve. A sufficient condition is given by a general result proved in [4]. For convenience, we define $\|Q\|_{L^\infty} := \max\{\|q_1\|_{L^\infty}, \|q_2\|_{L^\infty}\}$, for any $Q(t) = (q_1(t), q_2(t))$ piecewise continuous in $[a, b]$.

Theorem 4 *Let $P(t) = (x(t), y(t))$, $t \in [a, b]$, be a simple, piecewise C^1, and generalized regular curve, in the sense that it has no singular points (points where the left or right tangent vector is the zero vector) and no cusps (breakpoints where the left and right tangent vectors have opposite directions).*

Then, any piecewise C^1 closed approximating curve (with the same breakpoints), $\tilde{P}(t) = (\tilde{x}(t), \tilde{y}(t))$, $t \in [a, b]$, is simple itself, provided that the error

$$\|P - \tilde{P}\|_{PC^1} := \max\left\{\|P - \tilde{P}\|_{L^\infty}, \|P' - \tilde{P}'\|_{L^\infty}\right\}$$

is sufficiently small.

Proof. For the sake of concision, we recall only a qualitative proof, in the case that P and $\tilde{P}$ have no breakpoints and P is a regular curve (i.e., $P'(t^+) = P'(t^-) \neq (0, 0)$ for every $t \in (a, b)$ and $P'(a^+) = P'(b^-) \neq (0, 0)$), working by contradiction with some typical arguments of differential topology (cf., e.g., [7, Thm. 1.7]). The general proof is quite technical and resorts to the notion of generalized gradient of nonsmooth analysis; see [4], where also a quantitative proof is provided, with an estimate of the radius of a sufficient approximation neighborhood (even though not always simple to apply in practice).

Assume that the conclusion of the theorem is false. Then, there exists a sequence of C^1 curves, say $\{P_n\}$, with $\lim \|P_n - P\|_{C^1} = 0$, that are not simple, i.e., for every n there exist $u_n, v_n \in [a, b)$, or $u_n, v_n \in (a, b]$, $u_n \neq v_n$, such that $P_n(u_n) = P_n(v_n)$. By resorting possibly to subsequences, we may assume that $\lim u_n = u$ and $\lim v_n = v$ exist; since $\lim \|P_n - P\|_\infty = 0$, we have that $\lim P_n(u_n) = P(u) = P(v) = \lim P_n(v_n)$.

Now, we may have either $u = v$, or $u = b$ and $v = a$, or $u = a$ and $v = b$. Consider without loss of generality the case that either $u = v$ or $u = b$ and $v = a$, and define $\hat{v}_n = v_n$ if $v \neq a$, $\hat{v}_n = v_n + 1$ if $u = b$ and $v = a$ (i.e., $\lim \hat{v}_n = u$). Extend P (and P_n) to $[a, 2b - a]$ as $\hat{P}(t) = P(t)$, $t \in [a, b]$ and $\hat{P}(t) = P(t - (b - a))$, $t \in (b, b + (b - a)]$ (the extension being still C^1). Applying the Hermite-Genocchi formula to the first divided differences (cf., e.g., [2]), we can write

$$(0, 0) \equiv \frac{\hat{P}_n(u_n) - \hat{P}_n(\hat{v}_n)}{u_n - v_n} = \int_a^b \hat{P}_n'(tu_n + (1 - t)\hat{v}_n)\, dt$$

$$= \int_a^b \hat{P}'(tu_n + (1-t)\hat{v}_n)\, dt + E_n = \frac{\hat{P}(u_n) - \hat{P}(\hat{v}_n)}{u_n - \hat{v}_n} + E_n$$

where the vector sequence E_n tends to zero since $\|E_n\|_\infty \leq (b-a)\|P'_n - P'\|_\infty$. Now, we have assumed that $P(u)$ is not singular: taking the limit as $n \to \infty$ we get the contradiction $\hat{P}'(u) = P'(u) = (0,0)$. □

Remark 4 The kind of approximation in Theorem 4 is completely general. Indeed, it is only required that not only the curve, but also its tangent vectors are (piecewise) approximated. This means that the result can be applied for example to piecewise polynomial or trigonometric approximation, under suitable smoothness assumptions, and in general to any approximation process which guarantees convergence in PC^1 (the only constraint being that the approximating curve is closed if the original one is, a property that is guaranteed by any interpolation method including the endpoints of the parameter interval).

We recall that piecewise Chebyshev-Lobatto interpolation of maximal degree N guarantees convergence in PC^1 for functions that are piecewise $C^{3+\alpha}$, $\alpha > 0$, with order $\mathcal{O}(N^{-\alpha})$, in view of classical results concerning convergence of such process in Sobolev spaces; cf., e.g., [5, §5.5.3]. This gives a reasonable confidence that `Chebfun`, which approximates around machine precision, is able to produce a Jordan curve when the original one is a piecewise C^1 and generalized regular Jordan curve.

To conclude this section, it is worth discussing the following problem: can we give an error estimate like that of Theorem 3, in the case when the approximate boundary is not guaranteed to be a simple curve? This happens, for example, if the original curve has some singular point, so that Theorem 4 cannot be applied. Some classical closed parametric curves, like the cardioid, the tricuspoid, the nephroid, the bicorn, and many others, fall in this situation.

A partial answer can be given by the following result:

Theorem 5 *Let Ω be as in (15), and let $\tilde{P}(t) = (\tilde{x}(t), \tilde{y}(t))$, $t \in [a,b]$, be the piecewise polynomial approximating curve (16). Assume that the integrand f is Hölder-continuous with constant $\mathcal{C}$ and exponent $0 < \alpha \leq 1$ on the minimal rectangle $\mathcal{R} = [x_1, x_2] \times [y_1, y_2]$ containing $\Omega \cup \tilde{\Gamma}$, where $\tilde{\Gamma} = \{\tilde{P}(t)\,,\ t \in [a,b]\}$.*

Then, the following estimate holds for the error of the cubature formula (4) with $S = \tilde{P}$

$$|I_\Omega(f) - I_{2n-1}(f;\tilde{P})| \leq (x_2 - x_1)(\ell(\tilde{\Gamma}) + \ell_n)E_{2n-1}(f;\mathcal{R})$$

$$+(x_2 - x_1)\left(\ell(\partial\Omega)\,\mathcal{C}\,\varepsilon^\alpha + \|f\|_{\mathcal{R}}\,\|y' - \tilde{y}'\|_{L^1(a,b)}\right)$$

where $\lim \ell_n = \ell(\tilde{\Gamma})$.

Proof. Take $p^*_{2n-1} \in \mathbb{P}^2_{2n-1}$, for brevity p^*, such that $E_{2n-1}(f;\mathcal{R}) = \|f - p^*\|_{\mathcal{R}}$. Using the primitive (10) with $\xi \in [x_1, x_2]$, we can write the following chain of estimates

$$|I_\Omega(f) - I_{2n-1}(f;\tilde{P})| \leq \left| \oint_{\partial\Omega} \mathcal{F}_f\, dy - \oint_{\tilde{\Gamma}} \mathcal{F}_f\, dy \right| + \left| \oint_{\tilde{\Gamma}} \mathcal{F}_f\, dy - \oint_{\tilde{\Gamma}} \mathcal{F}_{p^*}\, dy \right|$$

$$+\left|\oint_{\tilde{\Gamma}} \mathcal{F}_{p^*}\, dy - I_{2n-1}(p^*;\tilde{P})\right| + |I_{2n-1}(p^*;\tilde{P}) - I_{2n-1}(f;\tilde{P})|$$

$$\leq \left(\int_a^b \left|\mathcal{F}_f(P(t)) - \mathcal{F}_f(\tilde{P}(t))\right| |y'(t)|\, dt + \int_a^b \left|\mathcal{F}_f(\tilde{P}(t))\right| |y'(t) - \tilde{y}'(t)|\, dt\right)$$

$$+\int_a^b \left|\mathcal{F}_{f-p^*}(\tilde{P}(t))\right| |\tilde{y}'(t)|\, dt + \sum_{i=1}^{L}\sum_{j=1}^{n}\sum_{k=1}^{n_i} |w_{ijk}|\, |p^*(x_{ijk}, y_{ik}) - f(x_{ijk}, y_{ik})|$$

where we have used the fact that $\oint_{\tilde{\Gamma}} \mathcal{F}_{p^*}\, dy = I_{2n-1}(p^*;\tilde{P})$ by construction of the cubature formula (4). Now, observing that for every $Q = (q_1, q_2) \in \Omega \cup \tilde{\Gamma} \subseteq \mathcal{R}$ and for every function g continuous in $\mathcal{R}$

$$|\mathcal{F}_g(Q)| \leq \int_{\xi}^{q_1} |g(s, q_2)|\, ds \leq |q_1 - \xi|\, \|g\|_{\mathcal{R}} \leq (x_2 - x_1)\, \|g\|_{\mathcal{R}}$$

and using the Hölder continuity of f and Theorem 2, we get easily the final cubature error estimate. □

3 Numerical tests

In this section we present a brief description of some relevant software features, and several numerical tests.

The routine `ChebfunGauss` (cf. [17]) implements the cubature formula defined above and is based on the `Chebfun` system (that requires at least a Matlab version 7.4). Given the boundary $\partial\Omega$ by two univariate functions $(x(t), y(t))$, $t \in [a, b]$, we substitute them by the piecewise polynomials $(\tilde{x}, \tilde{y})$ computed by `Chebfun`. For example, by

```
x_tilde=chebfun('cos(t)',[0 2*pi]);
y_tilde=chebfun('sin(t)',[0 2*pi]);
```

we obtain the approximation of the boundary of the unit disk. When the functions are only piecewise smooth, `Chebfun` is able to detect their singularities. Next, it is necessary to compute the derivative of $\tilde{x}$ simply by applying the Chebfun command `diff` to `x_tilde`. Once we have achieved the endpoints (singularities) of $\tilde{x}, \tilde{y}$ by

```
x_tilde_endpoints=x_tilde.ends;
y_tilde_endpoints=y_tilde.ends;
```

and the degrees of the piecewise polynomials by

```
x_tilde_degrees=x_tilde.funs;
y_tilde_degrees=y_tilde.funs;
```

we can easily get the breakpoints V_i and the local polynomials $S_{i,1} \in \mathbb{P}^2_{i_1}$, $S_{i,2} \in \mathbb{P}^2_{i_2}$, and build the cubature formula as described in Theorem 1 (by a suitable modification of the routine `SplineGauss` (cf. [16])).

In the numerical experiments, we have integrated five test functions over five domains with (piecewise) smooth boundary, namely the unit disk, a lune, the union and the intersection of two disks, a cardioid, and a deltoid, with different algebraic degrees of exactness (shortened ADE). As test functions we have chosen

$$
\begin{aligned}
f_1(x,y) &= (x+y)^{19} \\
f_2(x,y) &= \exp\left(-((x-x_0)^2+(y-y_0)^2)\right) \\
f_3(x,y) &= \exp\left(-100((x-x_0)^2+(y-y_0)^2)\right) \\
f_4(x,y) &= \sqrt{(x-x_0)^2+(y-y_0)^2} \\
f_5(x,y) &= \cos\left(20(x+y)\right) \qquad (22)
\end{aligned}
$$

namely a bivariate polynomial of degree 19, two gaussians with different variance parameter, an euclidean distance function, and an oscillating function. We have chosen an internal point (x_0, y_0) for all the domains of the examples below, namely $(x_0, y_0) = (0.5, 0.5)$ in Examples 3.1-3.5 and $(x_0, y_0) = (0, 0)$ in Example 3.6. The reference values of the integrals have been computed by a combination of methods, including the use of our formula at very high degree of exactness.

When the domains satisfy Property N, this is not inherited, in general, after boundary approximation. Nevertheless, in all the corresponding tests the cubature nodes turned out to be in the domain and the weights to be positive. This not surprising, in view of the fact that such are the nodes and weights if we use directly $x(t)$ and $y(t)$ instead of the approximants $S_{i,j}(t)$ in (5)-(6), and that the approximation errors are not far from machine precision.

The purpose of the numerical tests, is to show the capability of the numerical code `ChebfunGauss` to find automatically singularities and approximate accurately the boundaries (due to the features of `Chebfun`), and then to compute the integral using, at the same error level, less points than the Matlab's built-in `dblquad` and the spline-based code `SplineGauss` [16]. Moreover, `ChebfunGauss` turns out to be competitive also with the adaptive vectorized cubature code `TwoD` (cf. [13]), especially with regular integrands. In all the examples, the most difficult function to integrate is f_4, due to the singularities appearing in the derivatives at the internal point (x_0, y_0).

3.1 Disk

On the unit disk, with boundary parametrized as

$$P(t) = (\cos(t), \sin(t)) \ , \quad t \in [0, 2\pi]$$

we have chosen as base-line l the y-axis (see Property N); the boundary is approximated around machine precision by `Chebfun` with $\deg(\tilde{x}) = 20$ and $\deg(\tilde{y}) = 21$. The reference integrals are $I_\Omega(f_1) = 0$, $I_\Omega(f_2) = 1.476139002266508$, $I_\Omega(f_3) = 0.03141527827120767$, $I_\Omega(f_4) = 2.8540799175110$, $I_\Omega(f_5) = 0.02301872$ 593651162.

The error assumption of Theorem 4 is satisfied using a `Chebfun` representation of the circle, since, as it has been shown in [4], any C^1 curve approximating the circle is simple provided that $\|P - \tilde{P}\|_{C^1} < \sqrt{4\pi^2 + 8} - 2\pi) = 0.607...$.

We observe that in the column of function f_1 in Table 1 we have displayed the absolute error (and not the relative one as in the other examples) since the exact value of the integral is 0 (as it is trivial by symmetry). In Figure 1 we show the cubature points for ADE = 11.

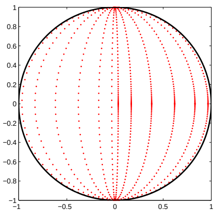

Figure 1: Cubature points on the unit disk for ADE = 11 with base-line $x = 0$.

Of course, one can observe that taylored rules for the disk use much less points at the same error level, but one has to keep in mind the flexibility of our routine `ChebfunGauss`, which can work on quite general domains, as it will be shown by the next examples.

Table 1: Cubature relative errors on the disk.

ADE	f_1	f_2	f_3	f_4	f_5
11	$9.5E-14$	$4.1E-11$	$4.2E-02$	$3.0E-05$	$2.5E-02$
21	$3.8E-14$	$2.4E-15$	$1.7E-03$	$2.1E-05$	$5.4E-08$
31	$3.4E-13$	$2.7E-15$	$5.2E-06$	$3.4E-06$	$4.1E-14$
41	$9.8E-14$	$3.2E-15$	$4.1E-09$	$3.8E-06$	$1.6E-13$

3.2 Lune

We consider a *lune* Ω defined as the difference of two disks with radius 0.5 centered in $(0.5, 0.5)$ and $(0, 0)$, respectively. The boundary $\partial\Omega$ can be represented by the curve

$$P(t) = \begin{cases} 0.5(1+\cos(t), 1+\sin(t)) \ , \quad t \in [-\pi/2, \pi] \\ \\ 0.5(\cos(3\pi/2 - t), \sin(3\pi/2 - t)) \ , \quad t \in [\pi, 3\pi/2] \end{cases}$$

In order to show the influence of the choice of the base-line l and of the resulting distribution of cubature nodes, we have chosen two different base-lines, namely $x = 0$ (Property N is not satisfied, some nodes lie outside the integration domain, cf. Fig. 2 and Table 2), and $x = 0.5$ (Property N is satisfied, all nodes belong to the integration domain, cf. Fig. 3 and Table 3). The boundary is approximated around machine precision by `Chebfun` with $\deg(\tilde{x}) = \deg(\tilde{y}) = 19$ on the first arc, and $\deg(\tilde{x}) = \deg(\tilde{y}) = 19$ on the second arc. It is worth observing that `Chebfun` automatically detects the breakpoint of the boundary at $(0, 0.5)$, whereas this is not necessary for $(0.5, 0)$ since it is used as start and end point of the parametrization. The reference integrals are $I_\Omega(f_1) = 638.5574327469890$, $I_\Omega(f_2) = 0.5726372043252941$, $I_\Omega(f_3) = 0.03137185199245524$, $I_\Omega(f_4) = 0.2064677029709676$, $I_\Omega(f_5) = 0.006289581219565822$. The curve is generalized regular, since the breakpoints $(0.5, 0)$ and $(0, 0.5)$ are not cusps.

This example was considered also in [15]. In Table 4 we compare the number of function evaluations needed by Matlab `dblquad` and by the adaptive vectorized code `TwoD` by Shampine [13], with those of `SplineGauss` and `ChebfunGauss`, at the same level of accuracy. Such accuracies have been taken from [15, Table 5], where the boundary of the lune is approximated by quintic splines with 128 interpolation points (65 points on each of the two circular arcs). In the case of `ChebfunGauss` they were obtained by a rule having degree of exactness 13, 9, 21, 15, 17, and show the better performance of this method, which uses about 5 percent of the number of points w.r.t. `SplineGauss`, and from less than 1 to 20 percent w.r.t. `dblquad` depending on the regularity of the function. `ChebfunGauss` is also competitive with `TwoD` (where we have used the '`singular`' option), except for the peaked function f_3 and for f_4 which has singularities of the derivatives at the internal point $(0.5, 0.5)$.

Table 2: Cubature relative errors on the lune, cf. Fig. 2.

ADE	f_1	f_2	f_3	f_4	f_5
11	$5.3E-06$	$8.6E-11$	$3.6E-01$	$3.2E-03$	$1.3E-01$
21	$5.0E-14$	$1.4E-15$	$1.2E-02$	$7.5E-04$	$2.2E-06$
31	$5.1E-14$	$1.4E-15$	$8.9E-05$	$1.9E-04$	$6.0E-14$
41	$4.5E-14$	$9.7E-16$	$2.0E-07$	$1.1E-04$	$9.0E-14$

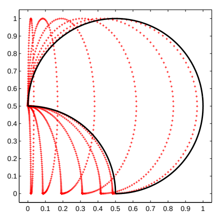

Figure 2: Cubature points on the lune for ADE $= 11$ with base-line $x = 0$.

Table 3: Cubature relative errors on the lune, cf. Fig. 3.

ADE	f_1	f_2	f_3	f_4	f_5
11	$1.7E-09$	$2.7E-15$	$1.3E-03$	$6.1E-06$	$3.9E-04$
21	$4.8E-14$	$0.0E+00$	$3.4E-08$	$1.3E-07$	$1.0E-12$
31	$5.0E-14$	$1.9E-16$	$9.5E-13$	$1.4E-07$	$2.9E-14$
41	$4.2E-14$	$1.6E-15$	$4.9E-14$	$2.4E-08$	$1.7E-14$

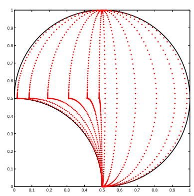

Figure 3: Cubature points on the lune for ADE $= 11$ with base-line $x = 0.5$.

Table 4: Number of cubature points used at comparable accuracies by the Matlab standard `dblquad` and Shampine's `TwoD` integrators, and by our formulae `SplineGauss`, `ChebfunGauss`.

	f_1	f_2	f_3	f_4	f_5
accuracy	$5E-11$	$6E-11$	$3E-08$	$5E-07$	$7E-10$
`dblquad`	282738	225950	20782	32740	1007426
`TwoD`	7056	1764	2744	980	46256
`SplineGauss`	34048	17920	81664	67840	55296
`ChebfunGauss`	1687	885	4059	5629	2745

3.3 Union of two disks

The domain Ω is defined as the union of two disks with radius 1 centered in $(\sqrt{2}/2, 0)$ and $(-\sqrt{2}/2, 0)$, and its boundary can be represented by the curve

$$P(t) = \begin{cases} (\sqrt{2}/2 + \cos(t), \sin(t)) \ , \quad t \in [-3\pi/4, 3\pi/4] \\ \\ (-\sqrt{2}/2 + \cos(t-\pi/2), \sin(t-\pi/2)) \ , \quad t \in [3\pi/4, 9\pi/4] \end{cases}$$

which is generalized regular (see Theorem 4), since the two breakpoints are not cusps. The chosen base-line is the x-axis, which gives Property N to the (nonconvex) domain; the boundary is approximated around machine precision by `Chebfun` with $\deg(\tilde{x}) = 18$, $\deg(\tilde{y}) = 19$ on the first arc, and $\deg(\tilde{x}) = 18$, $\deg(\tilde{y}) = 19$ on the second arc. The reference integrals are $I_\Omega(f_1) = 1.48710002014055437 \cdot 10^{-10}$, $I_\Omega(f_2) = 2.03976660215986882$, $I_\Omega(f_3) = 0.03141592653403229$, $I_\Omega(f_4) = 6.528597310412948$, $I_\Omega(f_5) = -0.01231537275802433$. See Fig. 4 and Table 5.

Table 5: Cubature relative errors on the union of two disks.

ADE	f_1	f_2	f_3	f_4	f_5
11	$2.8E-11$	$4.5E-11$	$3.1E-01$	$9.2E-05$	$2.8E-02$
21	$3.9E-11$	$2.6E-15$	$8.5E-03$	$1.7E-05$	$3.0E-09$
31	$8.4E-11$	$4.8E-15$	$5.2E-05$	$4.5E-06$	$1.5E-14$
41	$1.3E-11$	$2.8E-15$	$9.5E-08$	$1.4E-06$	$8.2E-14$

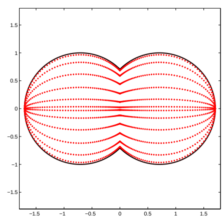

Figure 4: Cubature points on the union of two disks for ADE = 11 with base-line $y = 0$.

3.4 Intersection of two disks

The domain Ω is defined as the union of two disks with radius 1 centered in $((1+\sqrt{2})/2, 1/2)$ and $((1-\sqrt{2})/2, 1/2)$, and its boundary can be represented by the curve

$$P(t) = \begin{cases} ((1-\sqrt{2})/2 + \cos(t), 1/2 + \sin(t)) \ , & t \in [-\pi/4, \pi/4] \\ \\ ((1+\sqrt{2})/2 + \cos(t+\pi/2), 1/2 + \sin(t+\pi/2)) \ , & t \in [\pi/4, 3\pi/4] \end{cases}$$

which again is generalized regular (see Theorem 4). As base-line l we have chosen the straighline $x = 0.5$ joining the breakpoints $(1/2, (1-\sqrt{2})/2)$ and $(1/2, (1+\sqrt{2})/2)$, which gives Property N to the domain; the boundary is approximated around machine precision by `Chebfun` with $\deg(\tilde{x}) = 12$, $\deg(\tilde{y}) = 13$ on the first arc, and $\deg(\tilde{x}) = 12$, $\deg(\tilde{y}) = 13$ on the second arc. The reference integrals are $I_\Omega(f_1) = 457.0643824458973$, $I_\Omega(f_2) = 0.506809857736930103$, $I_\Omega(f_3) = 0.03141463000651704$, $I_\Omega(f_4) = 0.1827876057321204$, $I_\Omega(f_5) = 0.0049323361620 31037$. See Fig. 5 and Table 6.

Table 6: Cubature relative errors on the intersection of two disks.

ADE	f_1	f_2	f_3	f_4	f_5
11	$1.6E-12$	$2.8E-15$	$2.2E-05$	$4.5E-06$	$2.2E-08$
21	$3.4E-14$	$4.6E-15$	$1.6E-11$	$1.5E-06$	$2.4E-14$
31	$3.5E-14$	$3.7E-15$	$4.1E-14$	$3.9E-07$	$2.6E-14$
41	$3.0E-14$	$3.7E-15$	$3.1E-14$	$2.3E-07$	$5.2E-14$

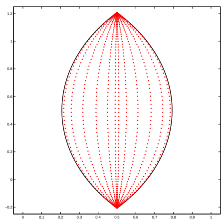

Figure 5: Cubature points on the intersection of two disks for ADE $= 11$ with base line $x = 0.5$.

3.5 Cardioid

The domain Ω is a cardioid, whose boundary has parametric equations

$$P(t) = ((1-\cos(t))\cos(t) + 1, (1-\cos(t))\sin(t)) \ , \quad t \in [0, 2\pi]$$

As base-line l we have chosen the straightline $x = 0.25$, which gives Property N to the domain; the boundary is approximated around machine precision by `Chebfun` with $\deg(\tilde{x}) = 26$ and $\deg(\tilde{y}) = 27$. Notice that the point $(1, 0)$ is singular, so that Theorems 3-4 cannot be invoked (the curve is not generalized regular). Nevertheless, the error estimates of Theorem 5 here apply. The reference integrals are $I_\Omega(f_1) = 22718.51704296164$, $I_\Omega(f_2) = 2.080016120389035$, $I_\Omega(f_3) = 0.03141592653589625$, $I_\Omega(f_4) = 4.547551380452429$, $I_\Omega(f_5) = 0.007169 671894735140$. See Fig. 6 and Table 7.

Table 7: Cubature relative errors on the cardioid.

ADE	f_1	f_2	f_3	f_4	f_5
11	$3.7E-07$	$1.8E-10$	$2.2E-01$	$9.5E-05$	$2.1E-03$
21	$4.2E-15$	$3.2E-15$	$1.6E-03$	$1.5E-05$	$5.1E-06$
31	$4.8E-16$	$4.7E-15$	$2.7E-06$	$3.0E-07$	$1.2E-12$
41	$1.1E-14$	$6.4E-16$	$5.6E-09$	$2.4E-06$	$3.9E-13$

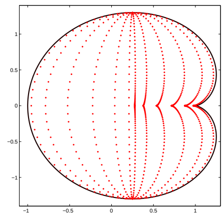

Figure 6: Cubature points on the cardioid for ADE = 11 with base-line $x = 0.25$.

3.6 Deltoid

The domain Ω is a deltoid, whose boundary is a tricuspoid (also known as Steiner's hypocicloid) with parametric equations

$$P(t) = a(2\cos(t) + \cos(2t), 2\sin(t) - \sin(2t)) \ , \quad t \in [0, 2\pi]$$

where we set $a = 1/3$ (cf. [8] for the properties of such a curve). Due to the shape of the boundary, there is no way to satisfy Property N, so we have chosen as base-line the straightline $y = 0$ (which is a symmetry axis of the domain); the boundary is approximated around machine precision by `Chebfun` with $\deg(\tilde{x}) = 26$ and $\deg(\tilde{y}) = 27$ on each of the three regular arcs. Due to the presence of three singular points, Theorems 3-4 cannot be invoked (the curve is not generalized regular). Nevertheless, the error estimates of Theorem 5 here apply. The reference integrals are $I_\Omega(f_1) = -0.171954131235978608$, $I_\Omega(f_2) = 0.597965094725642077$, $I_\Omega(f_3) = 0.0314158189238754187$, $I_\Omega(f_4) = 0.254933641266777200$, $I_\Omega(f_5) = 0.0167715230924418007$. See Fig. 7 and Table 8.

Table 8: Cubature relative errors on the deltoid, cf. Fig. 7.

ADE	f_1	f_2	f_3	f_4	f_5
11	$2.5E-05$	$3.2E-13$	$3.7E-04$	$6.1E-06$	$3.6E-02$
21	$7.4E-14$	$7.8E-15$	$4.3E-10$	$3.6E-06$	$2.2E-08$
31	$8.7E-14$	$4.8E-15$	$1.0E-13$	$5.5E-07$	$9.1E-14$
41	$2.5E-13$	$1.0E-14$	$1.6E-14$	$4.1E-07$	$3.0E-14$

References

[1] T.M. Apostol, *Calculus, Vol. II*, Blaisdell, Toronto, 1969.

[2] K.E. Atkinson, *An introduction to numerical analysis*. Second edition, John Wiley, 1989.

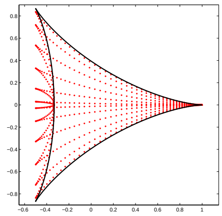

Figure 7: Cubature points on the deltoid for ADE = 11 with base-line $y = 0$.

[3] Z. Battles and L.N. Trefethen, *An extension of Matlab to continuous functions and operators*, SIAM J. Sci. Comp. 25 (2004), 1743–1770.

[4] L. Bos and M. Vianello, *On simple approximations to simple curves*, Dolomites Research Notes on Approximation 3 (2010), 1–6 (http://drna.di.univr.it).

[5] C. Canuto, M.Y. Hussaini, A. Quarteroni and T.A. Zang, *Spectral Methods (Fundamentals in Single Domains)*, Springer, 2006.

[6] R. Cools and A. Haegemans, *Algorithm 824: CUBPACK: A Package for Automatic Cubature; Framework Description*, ACM Trans. Math. Software 29 (2003), 287–296.

[7] M.W. Hirsch, *Differential Topology*, Springer, 1994.

[8] E.H. Lockwood, *A book of curves*, paperback re-issue of the 1963 edition, Cambridge University Press, Cambridge, 2007.

[9] The MathWorks, *MATLAB Documentation Set*, 2010 version (http://www.mathworks.com).

[10] R. Pachón, R. Platte and L.N. Trefethen, *Piecewise smooth chebfuns*, Technical Report NA-08/07, Oxford University Computing Laboratory, May 2008.

[11] R. Platte and L.N. Trefethen, *Chebfun: a new kind of numerical computing*, ECMI Proceedings, 2009 (http://www2.maths.ox.ac.uk/chebfun/publications).

[12] G. Santin, *Algebraic cubature on general domains by the software system Chebfun* (italian), Laurea Thesis in Mathematics, University of Padova, 2009 (advisors: A. Sommariva and M. Vianello).

[13] L.F. Shampine, *MATLAB program for quadrature in 2D*, Appl. Math. Comput. 202 (2008), 266–274.

[14] A. Sommariva and M. Vianello, *Product Gauss cubature over polygons based on Green's integration formula*, BIT Numerical Mathematics 47 (2007), 441–453.

[15] A. Sommariva and M. Vianello, *Gauss-Green cubature and moment computation over arbitrary geometries*, J. Comput. Appl. Math. 231 (2009), 886–896.

[16] A. Sommariva and M. Vianello, *SplineGauss: a Matlab code for Gauss-Green cubature over spline curvilinear polygons* (software available online at http://www.math.unipd.it/marcov/∼CAAsoft.html).

[17] A. Sommariva and M. Vianello, *ChebfunGauss: a Matlab code for Gauss-Green cubature by the Chebfun package* (software available online at http://www.math.unipd.it/marcov/∼CAAsoft.html).

[18] L.N. Trefethen and others, *Chebfun Version 4.0*, The Chebfun Development Team, 2011 (http://www.maths.ox.ac.uk/chebfun).