

ADAPTIVE QUADRATURES OVER SURFACES¹

KURT GEORG^{2,3} AND RALF WIDMANN^{4,5}

Abstract. Recently, the first author introduced a new approach for the numerical quadrature of surface integrals in the context of boundary element methods. The handling of singularities is an essential feature. It is assumed that a global parametrization \mathcal{P} of the surface is given as a subroutine and that \mathcal{P} is not accessible analytically. Of particular interest are parametrizations which are based on automatic triangulations of surfaces. The present paper improves and advances the method in various ways: The background is simplified and generalized, the parametrization is based on a recently developed efficient automatic triangulation of surfaces, an adaptive (recursive) quadrature method has been implemented which incorporates adaptive extrapolation steps. Several examples display the performance of the method. Codes (in C) can be obtained from the authors via e-mail.

Key words. Surface integrals, adaptive integration, piecewise linear surface approximation, extrapolation method, boundary element method, quadrature formula, trapezoidal rule.

AMS(MOS) subject classifications. 65D30, 65R20, 65N38

1. Introduction. Recently, Georg [18] introduced a new approach for the numerical quadrature of surface integrals. Interest there concentrated upon applications to boundary element methods. Hence the handling of singularities was essential. It was assumed that the surface was modeled via a piecewise linear approximation (triangulation). Such approximations are typically used in panel methods. Some error estimates can be found in [19].

Automatic triangulations of surfaces have been studied by Allgower, Georg, Gnutzmann, Rheinboldt, Schmidt, Widmann, see [2], [3], [5], [6], [7], [8], [21], [27], [29]. In particular, Widmann [29] implemented an efficient algorithm for triangulating surfaces in \mathbf{R}^3 .

The present paper extends and advances beyond [18] in the following respects:

1. The background is simplified and generalized, see Assumption 1 below. We incorporate the quadrature rule investigated in [18] into a versatile integration technique for general surfaces.
2. Our algorithm is based on the recently developed efficient automatic triangulation of surfaces [29].
3. An adaptive (recursive) procedure has been implemented which incorporates extrapolation steps.

An adaptive integration method for volume integrals based on a subdivision method in [30] and [4] has recently been given by the authors in [20]. However, the case of surfaces (without boundaries) is quite different and needs special numerical techniques.

¹ Preprint, Colorado State University, April 1993

² Partially supported by the National Science Foundation under grant number DMS-9104058

³ Department of Mathematics, Colorado State University, Ft. Collins, Colorado 80523, USA, e-mail: Georg@math.colostate.edu

⁴ Partially supported by a post-doctoral stipend of the Deutsche Forschungsgemeinschaft

⁵ Mathematisches Institut A, Universität Stuttgart, W-7000 Stuttgart 80, Fed. Rep. of Germany

We assume that the following is given:

ASSUMPTION 1. *Let $\mathcal{S} \subset \mathbf{R}^3$ be a piecewise smooth compact surface (with or without boundary) and $\mathcal{S}_{\text{PL}} \subset \mathbf{R}^3$ a piecewise linear compact surface which is assumed to approximate \mathcal{S} in the following sense: There is an isomorphism $\mathcal{P} : \mathcal{S}_{\text{PL}} \rightarrow \mathcal{S}$ which is piecewise smooth and maintains orientation.*

We view \mathcal{S}_{PL} as a parameter set and \mathcal{P} as a map generating a *global parametrization* of \mathcal{S} . For the applications we have in mind (e.g., boundary element methods), the surfaces \mathcal{S} and \mathcal{S}_{PL} generally will be closed, i.e., will not have a boundary, but for the purpose of this paper there is no need to make this restriction.

In principle, the parameter map \mathcal{P} can be any map which satisfies Assumption 1. In particular, an engineer or scientist may want to write a subroutine which implements a subdivision of a surface into pieces (panels), where each piece is parametrized in a different way. The assumptions on \mathcal{P} then essentially require that the different parametrizations are consistent, i.e., they match on the boundaries of the pieces.

In our implementations, we usually handle compact surfaces which are implicitly given by an equation $H(x) = 0$. For such situations, we have efficient software [29] which automatically triangulates the surface \mathcal{S} into a piecewise linear surface \mathcal{S}_{PL} . It is then possible to generate the parametrization \mathcal{P} via Newton iterations, see (8)–(11) below.

Typically, a numerical approximation of a surface integral would be based on the derivative of \mathcal{P} , thus reducing the surface integral to a two-dimensional integral over a triangle. To be more specific, if $\sigma \subset \mathbf{R}^3$ is a triangle, \mathcal{P} maps σ onto \mathcal{S} , and μ denotes the standard measure on \mathcal{S} (i.e., the surface element), then

$$(2) \quad \int_{\mathcal{P}\sigma} f(x) \mu(dx) = \int_{\sigma} f(\mathcal{P}y) \mathcal{P}^*\mu(dy),$$

$$(3) \quad \text{where } \mathcal{P}^*\mu(dy) = \|\partial_1\mathcal{P}(y) \times \partial_2\mathcal{P}(y)\| dy$$

corresponds to the usual definition of a surface integral as given in a calculus course. Here $\partial_1\mathcal{P}$ and $\partial_2\mathcal{P}$ denote the directional derivatives of \mathcal{P} with respect to a positively oriented orthogonal coordinate system $y = (y_1, y_2)$ of the span of σ .

The right-hand side of (2) is a two-dimensional integral in planar coordinates and can be approximated via known methods, see [11], [12], [13], [14], [16], [17], [22], [23], [24], [26]. Some of these methods also apply to singular integrals.

However, having cases in mind which occur in boundary element methods, we do not assume that the map \mathcal{P} can be easily differentiated. In fact, we want to allow the case that \mathcal{P} is only given via an iteration process or in the form of some other subroutine. Hence, we would have to use numerical approximations of the directional derivatives. This approach is investigated by Atkinson [9].

In contrast to this procedure, we investigate an approach proposed in [18], i.e., to directly approximate the integral (2) without making use of the derivative of \mathcal{P} . Our basic integration formula is simple and similar to (but different than) the trapezoidal rule, see Formula (13) below. This rule is then employed in a composite fashion. Assuming (as a conjecture) that this composite quadrature rule can be expanded in terms of h^2 ,

where h is the meshsize of the subdivision, extrapolation steps are performed. However, since we want to cover the case of weakly singular integrands, these extrapolation steps are implemented in a cautious and adaptive fashion.

In Section 2, we briefly introduce the piecewise linear approximation technique on which our method is based. In Section 3, our adaptive extrapolation method is described. In Section 4, several numerical examples, involving smooth or piecewise smooth surfaces and smooth or weakly singular integrands, illustrate the performance of the proposed method.

Codes (in C) can be obtained from the authors via e-mail by writing to:

Georg@math.colostate.edu

2. Piecewise Linear Approximations of Surfaces. We begin by specifying the assumptions on \mathcal{S}_{PL} .

DEFINITION 4. *Let \mathcal{S}_{PL} be a finite set of triangles in \mathbf{R}^3 . We call \mathcal{S}_{PL} a piecewise linear compact surface if the following two assumptions hold:*

1. *Two different triangles meet only in a common edge or vertex or do not meet at all.*
2. *Each edge belongs to at most two triangles.*

The boundary of \mathcal{S}_{PL} consists of those edges which belong to only one triangle.

A boundary element approach of a three-dimensional PDE typically generates a piecewise linear compact surface without boundary as an approximation of the given surface (without boundary) in a first step. The triangles are also called panels. Sometimes, more general panels are permitted, but the above definition captures the essential features.

We are interested in particular in handling surfaces \mathcal{S} which are implicitly defined via an equation

$$(5) \quad H(x) = 0$$

where $H : \mathbf{R}^3 \rightarrow \mathbf{R}$ is a smooth or piecewise smooth and continuous function. Let us briefly sketch how \mathcal{S} can be triangulated in this case. For more details, we refer to [3], [5], [7], [8], [21], [29]. We will illustrate the method for the example of the Coxeter-Freudenthal triangulation [10], [15], since it is the simplest to describe.

If $\tau = [v_0, v_1, v_2, v_3]$ is a tetrahedron in \mathbf{R}^3 , then a vertex v_i is pivoted via the step $\bar{v}_i \leftarrow v_{i-1} - v_i + v_{i+1}$ where the subindices are taken modulo 3. Hereafter, $[v_0, v_1, \dots, v_k]$ denotes the k -dimensional simplex with vertices v_0, v_1, \dots, v_k .

If two tetrahedra are connected via a pivoting step, then we say that the two tetrahedra are adjacent. The smallest family \mathcal{T} of tetrahedra which contains τ and is closed under pivoting (adjacency) is in fact a triangulation of \mathbf{R}^3 , see [1].

Let $H_\tau : \mathbf{R}^3 \rightarrow \mathbf{R}$ be the affine function which coincides with H on the vertices of a tetrahedron τ . We call τ transversal if the equation

$$(6) \quad H_\tau(x) = 0$$

has a solution $x \in \tau$. To simplify the discussion, we make the regularity assumption that H does not vanish on any vertex of \mathcal{T} . However, non-regularity can easily be overcome, see, e.g., [3].

If τ is transversal, then $H_\tau^{-1}(0) \cap \tau$ consists either of a triangle or a quadrilateral. In the latter case, we subdivide $H_\tau^{-1}(0) \cap \tau$ into two triangles. The resulting family of triangles is a piecewise linear surface \mathcal{S}_{PL} which approximates the surface $\mathcal{S} = H^{-1}(0)$. The piecewise linear surface \mathcal{S}_{PL} is compact if \mathcal{S} is compact. Furthermore, neither surface has a boundary. The quality of the approximation is discussed in [2].

The above remarks show that an algorithm for generating \mathcal{S}_{PL} in essence only has to generate all transversal tetrahedra. If \mathcal{S}_{PL} is compact and connected, then the following gives a rough sketch of the method:

```

ALGORITHM 7.  $\mathcal{T} \leftarrow \text{generate\_transversal\_tetrahedra}(\tau)$ 
  %  $\tau$ : initial transversal tetrahedron
  %  $\mathcal{T}$ : list of transversal tetrahedra

   $\mathcal{T} \leftarrow \emptyset$ 
   $\Sigma \leftarrow \{\tau\}$            % list to be checked
  WHILE  $\Sigma \neq \emptyset$ 
    get  $\tau \in \Sigma$ 
    FOR all transversal  $\tilde{\tau}$  adjacent to  $\tau$ 
      IF  $\tilde{\tau} \notin \Sigma \cup \mathcal{T}$ 
        add  $\tilde{\tau}$  to  $\Sigma$ 
      END IF
    END FOR
    drop  $\tau$  from  $\Sigma$  and add it to  $\mathcal{T}$ 
  END WHILE

```

If \mathcal{S}_{PL} is not connected, then an initial transversal tetrahedron is needed from each connected component.

A recent implementation of the above algorithm which also incorporates other features has been given by Widmann [29]. Special emphasis is put on consistent orientation of \mathcal{S}_{PL} , computational efficiency and on possibilities to improve and uniformize the mesh of \mathcal{S}_{PL} . The resulting algorithm has been used in our numerical examples.

A global parametrization $\mathcal{P} : \mathcal{S}_{\text{PL}} \rightarrow \mathcal{S}$ can be generated via one of the following Newton-type methods

$$(8) \quad x_{i+1} = x_i - \frac{\nabla H(x_i)}{\|\nabla H(x_i)\|^2} H(x_i),$$

$$(9) \quad x_{i+1} = x_i - \frac{\nabla H(x_0)}{\|\nabla H(x_0)\|^2} H(x_i),$$

$$(10) \quad x_{i+1} = x_i - \frac{a(x_0)}{a(x_0) \cdot \nabla H(x_i)} H(x_i),$$

by setting

$$(11) \quad x_0 \in \mathcal{S}_{\text{PL}} \longmapsto \mathcal{P}(x_0) := \lim_{i \rightarrow \infty} x_i \in \mathcal{S}.$$

The third iteration (10) along a fixed direction $a(x_0)$ is most useful for the case of a nonsmooth (i.e., only piecewise smooth) surface. In this case, we have to use a *smoothed gradient* $a(x)$ in order to retain the global character of the parametrization, see the example in Section 4.4.

Many other iterations are possible; for example, it might be more efficient to use secant-type methods.

3. The Adaptive Extrapolation Method. Let $\mathcal{P} : \mathcal{S}_{\text{PL}} \rightarrow \mathcal{S}$ be a global parametrization of the compact surface \mathcal{S} . Our numerical examples in Section 4 are based on the methods described in the previous section, but we can allow any parametrization which satisfies Assumption 1.

We consider the surface integral

$$(12) \quad \int_{\mathcal{S}} f(x) \mu(dx) = \sum_{\sigma \in \mathcal{S}_{\text{PL}}} \int_{\mathcal{P}\sigma} f(x) \mu(dx)$$

which splits into a sum via the decomposition of the parameter set \mathcal{S}_{PL} into triangles. We emphasize that the generation of a suitable triangulation and global parametrization is a nontrivial initial step for approximating surface integrals, since it enables us to reduce the problem to the more elementary surface integrals (2).

The following basic quadrature formula for (2) was proposed in [18]: If $\sigma \in \mathcal{S}_{\text{PL}}$ has vertices v_1, v_2, v_3 , then a coarse approximation of the surface integral is given by

$$(13) \quad \int_{\mathcal{P}\sigma} f(x) \mu(dx) \approx \mathcal{I}(f, \sigma, \mathcal{P}) := \text{mean}_{i=1}^3 f(\mathcal{P}v_i) \text{area}_{i=1}^3 \mathcal{P}v_i,$$

where $\text{area}_{i=1}^3 \mathcal{P}v_i$ indicates the area of the triangle composed of the vertices $\mathcal{P}v_i$, and $\text{mean}_{i=1}^3 f(\mathcal{P}v_i)$ the mean value of the numbers $f(\mathcal{P}v_i)$. Note that this rule differs from the standard trapezoidal rule

$$(14) \quad \int_{\mathcal{P}\sigma} f(x) \mu(dx) \approx \text{mean}_{i=1}^3 f(\mathcal{P}v_i) \gamma(v_i) \text{area}_{i=1}^3 v_i$$

where

$$\gamma(v) = \|\partial_1 \mathcal{P}(v) \times \partial_2 \mathcal{P}(v)\|,$$

see Equation (3).

The quadrature formula (13) can be utilized in the following way: We subdivide each edge of the triangle σ into n equidistant parts. By connecting corresponding gridpoints on different edges, we generate n^2 small triangles $\mathcal{D}_n(\sigma)$. This leads to a composite rule. In [18], the following asymptotic expansion of this composite quadrature rule was conjectured for suitable smoothness assumptions:

$$(15) \quad \mathcal{I}_n(f, \sigma, \mathcal{P}) := \sum_{\rho \in \mathcal{D}_n(\sigma)} \mathcal{I}(f, \rho, \mathcal{P})$$

$$(16) \quad = \int_{\mathcal{P}\sigma} f(x) \mu(dx) + \frac{C_1}{n^2} + \frac{C_2}{n^4} + \dots + \frac{C_k}{n^{2k}} + \mathcal{O}\left(\frac{1}{n^{2k+2}}\right).$$

Unfortunately, we have as yet been unable to prove this conjecture, but we have convincing numerical evidence that it holds. A proof of (16) has been given by Lyness [25] for the different basic formula (14), but this proof which is based on the Euler–Maclaurin sum formula does not easily carry over to our case (formula (13)).

We now consider the process of successive halving based on (16), and use a standard Romberg extrapolation tableau:

$$(17) \quad \begin{array}{rcccc} \mathcal{I}_1(f, \sigma, \mathcal{P}) & = & T_{0,0} & & \\ \mathcal{I}_2(f, \sigma, \mathcal{P}) & = & T_{1,0} & T_{1,1} & \\ \mathcal{I}_4(f, \sigma, \mathcal{P}) & = & T_{2,0} & T_{2,1} & T_{2,2} \\ & & \vdots & \vdots & \vdots & \ddots \\ \mathcal{I}_{2^n}(f, \sigma, \mathcal{P}) & = & T_{n,0} & T_{n,1} & T_{n,2} & \cdots & T_{n,n}. \end{array}$$

This method has been tested and seems to confirm the asymptotic behavior we anticipate for sufficiently smooth data. However, having applications to boundary element methods in mind, we have to allow for weak singularities in the integrand f , and then the asymptotic expansion (16) is not valid any more. Therefore we monitor the Romberg extrapolation tableau by tests. The essential idea of this test is to check whether the relation

$$(18) \quad \frac{T_{i-1,k} - T_{n,n}}{T_{i,k} - T_{n,n}} \approx 4^{k+1}$$

holds. We sketch the essential features of our method in the following recursive

ALGORITHM 19. Int \leftarrow adaptive_integr(Int, σ , Tol, n_{\max})

% Tol: tolerance

% n_{max}: maximal size of the extrapolation tableau (17)

% Int: recursively collects the terms of the integration, initially Int \leftarrow 0,

FOR $n = 2$ to n_{\max}

complete tableau (17)

IF the coefficient check (18) is positive

IF $|T_{n,n-1} - T_{n,n}| \leq \text{Tol}$

Int \leftarrow Int + $T_{n,n}$

EXIT adaptive_integr

END IF

ELSE

IF $|T_{n-1,0} - T_{n,0}| \leq \text{Tol}$

Int \leftarrow Int + $T_{n,0}$

EXIT adaptive_integr

ELSE *% recursion to avoid singularities*

split σ into four subtriangles τ_i , $i = 1, \dots, 4$

Int \leftarrow adaptive_integr(Int, τ_i , Tol, n_{\max}), $i = 1, \dots, 4$

END IF

END IF

END FOR

% recursion (case $n > n_{\max}$) to reduce the size of the extrapolation tableau (17)
 split σ into four subtriangles τ_i , $i = 1, \dots, 4$
 Int \leftarrow adaptive_integr(Int, τ_i , Tol, n_{\max}), $i = 1, \dots, 4$

Our implementation generates a rather complex tree structure in order to avoid repetitions of calculations and of function calls. In particular, the complexity of this tree grows rather rapidly with increasing size of the extrapolation tableau (17). It is therefore advisable to set n_{\max} to a moderate value.

We note in passing that the extrapolation steps can probably be improved by considering rational function extrapolation and the Bulirsch numbers 2, 4, 6, 8, 12, 16, \dots , see Stoer and Bulirsch [28, Section 7.2.12].

4. Numerical Examples. Let us consider some numerical examples in order to illustrate the features of our integration method. The tables below show the performance of the method for different tolerances “Tol” and list the approximate value for the integral under “Integral”. Furthermore, the “Relative Error” is either calculated (for the case that the exact integral is known) or is estimated (by using the best known approximation). Under “Time”, we list the run time of the integration. The program is written in C and was run on a PC with the 80386/387 processors.

4.1. First Example: Smooth Surface, Smooth Integrand. We choose the integrand

$$(20) \quad f(x) = 1$$

and the surface \mathcal{S} to be one eighth of the unit sphere. We set $\mathcal{S}_{\text{PL}} = \{[e_1, e_2, e_3]\}$, where e_i denote the standard unit basis vectors. Clearly, a valid parametrization is obtained via $\mathcal{P}x = x/\|x\|_2$. However, to be consistent with more complex situations where an explicit parametrization will be difficult to obtain, we generate \mathcal{P} via the iteration (8) with $H(x) = \|x\|_2^2 - 1$. Thus, we obtain a more realistic example where the iteration which generates \mathcal{P} contributes to the computational complexity of the method.

The maximal size n_{\max} of the extrapolation tableau (17) is set to $n_{\max} = 3$. With a maximal size of 4 or 5, the results for Tol = $10^{-2} - 10^{-8}$ remain unchanged, for higher tolerances the computation time increases a bit but the result does not become more accurate.

TABLE 21. Integrand (20), \mathcal{S} = one eighth of the sphere, $\mathcal{S}_{\text{PL}} = [e_1, e_2, e_3]$, the exact value is $\pi/2$.

Tol	Integral	Rel. Error	Time
10^{-2}	1.5703621925192	2.8e-04	0.06
10^{-3}	1.5703621925192	2.8e-04	0.06
10^{-4}	1.5707895985010	4.3e-06	0.38
10^{-5}	1.5707904092622	3.8e-06	0.44
10^{-6}	1.5707956530017	4.3e-07	1.34
10^{-7}	1.5707962503211	4.9e-08	2.42
10^{-8}	1.5707963204981	4.0e-09	5.02
10^{-9}	1.5707963258933	5.7e-10	6.45
10^{-10}	1.5707963264580	2.1e-10	7.85
10^{-11}	1.5707963267156	5.0e-11	24.96
10^{-12}	1.5707963267947	1.3e-13	34.36

4.2. Second Example: Smooth Surface, Singular Integrand. We choose the integrand

$$(22) \quad f(x) = \frac{\nu(x) \cdot (x - a)}{\|x - a\|^2},$$

where $\nu(x)$ is the outer normal vector of \mathcal{S} at x , and $a \in \mathcal{S}$. Here, we use $a = e_1$. The singularity in a is not strong; in fact, if \mathcal{S} is any surface which is smooth in a neighborhood of $a \in \mathcal{S}$, then

$$\limsup_{\substack{x \rightarrow a \\ x \in \mathcal{S}}} |f(x)| < \infty.$$

More precisely, if $x(\epsilon) \rightarrow a$ as $\epsilon \rightarrow 0$ on the surface $\mathcal{S} = H^{-1}(0)$, then

$$\lim_{\epsilon \rightarrow 0} f(x(\epsilon)) = \frac{H''(a)(\dot{x}(0), \dot{x}(0))}{2\|H'(a)\| \|\dot{x}(0)\|^2}.$$

TABLE 23. Integrand (22), \mathcal{S} = one eighth of the sphere, $\mathcal{S}_{\text{PL}} = [e_1, e_2, e_3]$, the exact value is $\pi/4$.

Tol	Integral	rel. Error	Time
10^{-2}	0.78451628790578	1.1e-03	0.24
10^{-3}	0.78433138851981	1.4e-03	0.33
10^{-4}	0.78527164978728	1.6e-04	0.63
10^{-5}	0.78539213336955	7.7e-06	0.93
10^{-6}	0.78539748890823	8.6e-07	2.03
10^{-7}	0.78539804139239	1.6e-07	2.39
10^{-8}	0.78539814552427	2.3e-08	5.65
10^{-9}	0.78539816060875	3.6e-09	6.70
10^{-10}	0.78539816319467	2.6e-10	9.42
10^{-11}	0.78539816334936	6.1e-11	23.18
10^{-12}	0.78539816339725	2.6e-13	37.07

The previous integrand was also investigated in the numerical tests of [18], where it was erroneously related to the solid angle. Instead, the solid angle is related to the next integrand

$$(24) \quad f(x) = \frac{\nu(x) \cdot (x - e_1)}{\|\nu(x)\| \|x - e_1\|^3}.$$

The error is calculated against the results obtained with TOL= 10^{-12} .

TABLE 25. Integrand (24), \mathcal{S} = one eighth of the sphere, $\mathcal{S}_{\text{PL}} = [e_1, e_2, e_3]$.

Tol	Integral	rel. Error	Time
10^{-2}	1.1047280500860	5.4e-03	0.39
10^{-3}	1.1078891108051	2.6e-03	0.49
10^{-4}	1.1100726604569	5.8e-04	0.94
10^{-5}	1.1106442261845	7.0e-05	2.20
10^{-6}	1.1107150936914	6.0e-06	6.76
10^{-7}	1.1107194840148	2.0e-06	18.73
10^{-8}	1.1107214101028	2.8e-07	45.09
10^{-9}	1.1107216846443	3.5e-08	112.10
10^{-10}	1.1107217213735	2.0e-09	227.07
10^{-11}	1.1107217239305	3.0e-10	418.31
10^{-12}	1.1107217235998	—	807.96

4.3. Third Example: ‘Nontrivial’ Smooth Surface, Singular Integrand.

In this example we integrate over the surface of a ring cyclide, i.e. a torus with varying radius. The surface is defined by the function

$$H(x_1, x_2, x_3) = (x_1^2 + x_2^2 + x_3^2 + R^2 - b^2 - k^2)^2 - 4(Rx_1 + kb)^2 - 4(R^2 - b^2)x_2^2 = 0.$$

Here, R defines the ‘big’ radius, and the small radius r satisfies $k - b \leq r \leq k + b$. We choose the parameters $R = 1, k = 0.3$ and $b = 0.15$. The piecewise linear surface \mathcal{S}_{PL} was generated with our software [29] using a mesh size of $\delta = 0.15$.

The integrand we use, is the one in equation (24) with e_1 replaced by $(1 + k + b)e_1$.

TABLE 26. Integrand (24), \mathcal{S} = ring cyclide, \mathcal{S}_{PL} consists of 4486 triangles, the exact value is 2π .

Tol	Integral	Rel. Error	Time
10^{-2}	6.2663343738077	2.7e-03	117.10
10^{-3}	6.2826505787538	8.5e-05	117.70
10^{-4}	6.2833390733353	2.4e-05	120.89
10^{-5}	6.2831439388331	6.6e-06	129.30
10^{-6}	6.2831590013010	4.2e-06	151.82
10^{-7}	6.2831695235343	2.5e-06	208.83
10^{-8}	6.2831855787218	4.3e-08	357.46
10^{-9}	6.2831855087918	3.2e-08	722.33
10^{-10}	6.2831853955353	1.4e-08	1363.09
10^{-11}	6.2831853924959	1.4e-08	3801.83
10^{-12}	6.2831853263395	3.0e-09	10872.55

Ring Cyclide

4.4. Fourth Example: Nonsmooth Surface, Singular Integrand. We consider the function

$$(27) \quad f(x) = \frac{1}{\sqrt{x_1^2 + x_2^2}},$$

which is singular on the x_3 -axis, and integrate it over the surface of two intersecting spheres

$$\mathcal{S} = \{x \in \mathbf{R}^3 \mid \min \{(x_1 - 0.75)^2, (x_1 + 0.75)^2\} + x_2^2 + x_3^2 - 1 = 0\}.$$

The global parametrization \mathcal{P} is generated by a one-dimensional Newton iteration along the direction of an exponentially smoothed gradient, see iteration (10).

It is possible to reduce the computation times by making use of the symmetries of the surface and the integrand. It is sufficient to calculate the integral on that part of the surface that lies in the first octant, i. e. $x, y, z \geq 0$, and multiply the result by 8.

For higher tolerances it turns out that a finer triangulation is advantageous.

TABLE 28. Integrand (27) (times 8), \mathcal{S} = one eighth of two intersecting spheres, \mathcal{S}_{PL} consists of 318 triangles.

Tol	Integral	Rel. Error	Time
10^{-2}	26.971577634897	6.2e-03	9.12
10^{-3}	27.087770186083	1.9e-03	9.17
10^{-4}	27.129786532562	3.3e-04	10.60
10^{-5}	27.138405437972	1.5e-05	13.02
10^{-6}	27.137645243439	4.3e-05	20.54
10^{-7}	27.138804397520	7.7e-07	37.48
10^{-8}	27.138823880163	5.1e-08	73.71
10^{-9}	27.138824808454	1.7e-08	140.63
10^{-10}	27.138825170085	3.4e-09	234.99
10^{-11}	27.138825247926	5.2e-10	394.26
10^{-12}	27.138825261714	—	698.21

REFERENCES

- [1] E. L. ALLGOWER AND K. GEORG, *Generation of triangulations by reflections*, Utilitas Math., 16 (1979), pp. 123–129.
- [2] ———, *Estimates for piecewise linear approximations of implicitly defined manifolds*, Appl. Math. Lett., 1 (1989), pp. 1–7.
- [3] ———, *Numerical Continuation Methods: An Introduction*, vol. 13 of Series in Computational Mathematics, Springer Verlag, Berlin, Heidelberg, New York, 1990. pp 388.
- [4] E. L. ALLGOWER, K. GEORG, AND R. WIDMANN, *Volume integrals for boundary element methods*, J. Comput. Appl. Math., 38 (1991), pp. 17–29.
- [5] E. L. ALLGOWER AND S. GNUTZMANN, *An algorithm for piecewise linear approximation of implicitly defined two-dimensional surfaces*, SIAM J. Numer. Anal., 24 (1987), pp. 452–469.
- [6] ———, *Simplicial pivoting for mesh generation of implicitly defined surfaces*, Computer Aided Geometric Design, 8 (1991), pp. 305–325.
- [7] E. L. ALLGOWER AND P. H. SCHMIDT, *An algorithm for piecewise-linear approximation of an implicitly defined manifold*, SIAM J. Numer. Anal., 22 (1985), pp. 322–346.
- [8] ———, *Computing volumes of polyhedra*, Math. Comp., 46 (1986), pp. 171–174.
- [9] K. E. ATKINSON, *Two-grid iteration method for linear integral equations of the second kind on piecewise smooth surfaces in \mathbf{R}^3* . Report 14, Univ. of Iowa, submitted to SIAM J. Numer. Anal., 1991.
- [10] H. S. M. COXETER, *Discrete groups generated by reflections*, Ann. of Math., 6 (1934), pp. 13–29.
- [11] P. J. DAVIS AND P. RABINOWITZ, *Methods of Numerical Integration*, Academic Press, Orlando, 1984. Second edition.
- [12] E. DE DONCKER, *An adaptive extrapolation algorithm for automatic integration*, SIGNUM Newsl, 13.2 (1978), pp. 12–18.
- [13] ———, *New Euler-Maclaurin expansions and their application to quadrature over the s-dimensional simplex*, Math. Comp., 33 (1979), pp. 1003–1018.
- [14] E. DE DONCKER AND I. ROBINSON, *Algorithm 612 - TRIEX, Integration over a TRIangle using nonlinear EXtrapolation*, ACM Trans. Math. Software, 10 (1984), pp. 17–22.
- [15] H. FREUDENTHAL, *Simplizialzerlegungen von beschränkter Flachheit*, Ann. of Math., 43 (1942), pp. 580–582.
- [16] J. H. FRIEDMAN AND M. H. WRIGHT, *A nested partitioning procedure for numerical multiple integration*, ACM Trans. Math. Software, 7 (1981), pp. 76–92.
- [17] F. N. FRITSCH, D. K. KAHANER, AND J. N. LYNNESS, *Double integration using one-dimensional adaptive quadrature routines: A software interface problem*, ACM Trans. Math. Software, 7 (1981), pp. 46–75.

- [18] K. GEORG, *Approximation of integrals for boundary element methods*, SIAM J. Sci. Stat. Comput., 12 (1991), pp. 443–453.
- [19] K. GEORG AND J. TAUSCH, *Some error estimates for the numerical approximation of surface integrals*. Preprint, Colorado State University, 1992.
- [20] K. GEORG AND R. WIDMANN, *Adaptive quadratures over volumes*, Computing, 47 (1991), pp. 121–136.
- [21] S. GNUTZMANN, *Stückweise lineare Approximation implizit definierter Mannigfaltigkeiten*, PhD thesis, University of Hamburg, Fed. Rep. Germany, 1989.
- [22] D. K. KAHANER AND O. W. RECHARD, *TWODQD an adaptive routine for two-dimensional integration*, J. Comput. Appl. Math., 17 (1987), pp. 215–234.
- [23] P. KEAST, *Cubature formulas for the surface of the sphere*, J. Comput. Appl. Math., 17 (1987), pp. 151–172.
- [24] J. N. LYNESS, *Applications of extrapolation techniques to multidimensional quadrature of some integrand functions with a singularity*, J. Comp. Phys., 20 (1976), pp. 346–364.
- [25] ———, *Quadrature over a simplex: Part 2. A representation for the error functional*, SIAM J. Numer. Anal., 15 (1978), pp. 870–887.
- [26] J. N. LYNESS AND B. J. J. MCHUGH, *Integration over multidimensional hypercubes. I: A progressive procedure*, Comput. J., 6 (1963), pp. 264–270.
- [27] W. C. RHEINBOLDT, *On a moving-frame algorithm and the triangulation of equilibrium manifolds*, in Bifurcation: Analysis, Algorithms, Applications, T. Küpper, R. Seydel, and H. Troger, eds., vol. 79 of ISNM, Basel, 1987, Birkhäuser Verlag, pp. 256–267.
- [28] J. STOER AND R. BULIRSCH, *Introduction to Numerical Analysis*, Springer Verlag, Berlin, Heidelberg, New York, 1980.
- [29] R. WIDMANN, *An efficient algorithm for the triangulation of surfaces in \mathbf{R}^3* . Preprint, Colorado State University, October 1990.
- [30] ———, *Efficient triangulation of 3-dimensional domains*. Preprint, Colorado State University, November 1990.