

APPROXIMATION OF INTEGRALS FOR BOUNDARY ELEMENT METHODS*

KURT GEORG†

Abstract. A new method for approximating two-dimensional integrals $\int_B f(x) \mu(dx)$ over surfaces $B \subset \mathbf{R}^3$ is introduced where μ is the standard measure of surface area. Such integrals typically occur in boundary element methods. The algorithm is based on triangulations $T := \bigcup T_i$ approximating B . Under the assumption that the surface B is given implicitly by an equation $H(x) = 0$, a retraction $P : U \supset B \rightarrow B$ is used to obtain a curved subdivision $B = \bigcup B_i$ via $B_i := PT_i$. Except in very special cases, this retraction is not analytically accessible, but is generated by a subroutine. Hence standard multiple integral techniques are not available. Thus, the approach given here differs from the usual panel method. It is shown how to calculate the integrals as precisely as wished. Two numerical examples are given. The first integrand $f(x) \equiv 1$ is regular, and it is shown that a very accurate extrapolation method can be used. The second integrand $f(x) \sim \|x - x_0\|^{-1}$ is singular, and an adaptive refinement procedure is displayed.

Key words. surface integration, quadrature formula, boundary element method, trapezoidal rule, extrapolation method, adaptive refinement

AMS(MOS) subject classifications. 65D30, 65N35, 65R20, 45E05

1. Motivation. Let us motivate the following discussion by considering a well-known integral equation related to Laplace's equation:

$$(1.1) \quad \begin{aligned} \Delta u &= 0 \quad \text{in } D, \\ u &= g \quad \text{on } B, \end{aligned}$$

where $D \subset \mathbf{R}^3$ is a bounded open nonempty domain with sufficiently regular boundary B . We introduce the fundamental solution

$$s(x, y) := \frac{1}{4\pi} \frac{1}{\|x - y\|},$$

and consider the following double layer potential ansatz for a solution u of (1.1):

$$(1.2) \quad u(x) = \int_B \frac{\partial s(x, y)}{\partial \nu(y)} \sigma(y) \mu(dy),$$

where $\nu(y)$ for $y \in B$ is the unit normal vector pointing out of D and σ is an unknown density function on B to be determined by the boundary integral method. Here and in the following, the symbol μ denotes the standard measure of surface area; $\mu(dy)$ is often called the surface element. It turns out, see, e.g., [9-12], that σ satisfies the following integral equation of the second kind:

$$(1.3) \quad \sigma(x) - 2 \int_B \frac{\partial s(x, y)}{\partial \nu(y)} \sigma(y) \mu(dy) = -2g(x).$$

*Received by the editors February 2, 1990; accepted for publication April 3, 1990. This work was supported by National Science Foundation grant DMS-8947761 and by Deutsche Forschungsgemeinschaft SFB 256.

†Department of Mathematics, Colorado State University, Fort Collins, Colorado 80523.

A typical discretization method for solving (1.3) is a collocation method, see, e.g., Hackbusch [11]. We introduce basis functions ϕ_j on B and corresponding collocation points $x_k \in B$ to obtain an interpolation of σ :

$$\sigma(x_k) = \sum_j \sigma_j \phi_j(x_k).$$

The following collocation equation determines the coefficients σ_j of this interpolation:

$$(1.4) \quad \sum_j \phi_j(x_k) \sigma_j - 2 \sum_j \int_B \frac{\partial s(x_k, y)}{\partial \nu(y)} \phi_j(y) \mu(dy) \sigma_j = -2g(x_k).$$

In order to solve the linear equation (1.4) for the unknowns σ_j , the matrix entries

$$(1.5) \quad M[k, j] := \int_B \frac{\partial s(x_k, y)}{\partial \nu(y)} \phi_j(y) \mu(dy)$$

have to be approximated. The generation of these or similar matrix entries is an essential topic in boundary element methods. Atkinson [4] discusses a Galerkin method for discretizing (1.1).

In order to find suitable basis functions for the above collocation method or for a Galerkin method, usually a triangulation T of the surface B is introduced. Since the surface is closed, i.e., has no boundary, such triangulations usually are obtained from a two-dimensional compact pseudomanifold \mathcal{T} without boundary, see, e.g., [2]. More precisely, we consider a finite collection

$$(1.6) \quad \mathcal{T} = \{T_i\}_{i=1, \dots, n}, \quad T := \bigcup_{i=1}^n T_i$$

of triangles T_i , each having affinely independent vertices in \mathbf{R}^3 , such that the following two conditions hold:

(1.7) Any two triangles $T_i \neq T_j$, $i \neq j$ intersect in a common edge or vertex or not at all.

(1.8) Any edge is common to exactly two different triangles of \mathcal{T} .

The panel methods that are very popular in engineering replace the surface B by the triangulation T . In our example for Poisson's equation (1.1), this replacement has to be performed in all integrals of (1.2), (1.3), (1.4), (1.5). Thus, a preliminary discretization error is generated which, for the purpose of the following discussion, will be called the *panel error*. This panel error has to be considered in addition to any discretization error which is generated by a collocation or Galerkin method. Unfortunately, the numerical analysis of the panel error is not clear at all, contrary to the discretization errors of collocation or Galerkin methods which are wellknown, see, e.g., [11]. At best, we can hope for a panel error of the order $O(h^2)$, where h measures the mesh size of the triangulation.

The advantage of the panel method is that basis functions ϕ_j can easily be obtained, similarly to the techniques in finite element methods. Typically, the support of such basis functions intersects only very few triangles of \mathcal{T} , so that the integrals which have to be computed in order to generate the matrix entries (1.5) can be taken over a few triangles. However, the panel error deprives us of the possibility of making use of

smoother elements. We now outline a different approach that still uses a triangulation but avoids the panel error. In addition, it is shown that the resulting integrals can be numerically approximated as precisely as wished.

2. A subdivision of the surface. We make use of the fact that the surface B is given implicitly by an equation. To be more precise, we make the following three assumptions:

(2.1) Let $H : \mathbf{R}^3 \rightarrow \mathbf{R}$ be smooth and 0 a regular value of H .

(2.2) $B := \{x \in \mathbf{R}^3 : H(x) = 0\}$.

(2.3) $D := \{x \in \mathbf{R}^3 : H(x) \leq 0\}$ is nonempty and bounded.

These conditions could be relaxed considerably, but this is not so important for the exposition of our ideas. We consider either of the following two Newton iterations for *orthogonally* converging towards the surface B :

(2.4a) **input** $x_0 \in \mathbf{R}^3$;
for $i = 0, 1, 2, \dots$ **do** $x_{i+1} \leftarrow x_i - \|\nabla H(x_i)\|^{-2} H(x_i) \nabla H(x_i)$;
if the iteration converges **then output** $\lim_{i \rightarrow \infty} x_i$.

(2.4b) **input** $x_0 \in \mathbf{R}^3$;
for $i = 0, 1, 2, \dots$ **do** $x_{i+1} \leftarrow x_i - \|\nabla H(x_0)\|^{-2} H(x_i) \nabla H(x_0)$;
if the iteration converges **then output** $\lim_{i \rightarrow \infty} x_i$.

It is clear from the local convergence of Newton's method and the assumptions (2.1)–(2.3) that there exists an open neighborhood U of B such that either one of the above input-output devices defines a smooth map $P : U \rightarrow B$, which is a retraction since $P(x) = x$ for $x \in B$. It is easy to implement this retraction numerically. This will be used in the sequel. Furthermore, for a given $\epsilon > 0$, it is also possible to reduce the neighborhood U so that $\|x - P(x)\| \leq \epsilon$ and $|\langle x - P(x), \nabla H(x) \rangle| \geq (1 - \epsilon)\|x - P(x)\|\|\nabla H(x)\|$ holds for all $x \in U$. Here $\langle \cdot, \cdot \rangle$ denotes the standard scalar product of \mathbf{R}^3 .

Intuitively, it is clear that a two-dimensional compact pseudomanifold \mathcal{T} without boundary, see (1.6), approximates B reasonably well if we can find a neighborhood U in the above sense and such that the following conditions hold:

(2.5) $T_i \in U$ for all $i = 1, \dots, n$.

(2.6) $\nabla H(x)$ is not tangent to T_i for all $x \in T_i$ and $i \in \{1, \dots, n\}$.

(2.7) The retraction P induces a topological isomorphism from T onto B .

It is possible to automatically generate triangulations via piecewise linear algorithms, see, e.g., [2,3,8] for a general reference. As was shown in [1], such triangulations have the above property for sufficiently small mesh sizes. In this paper, we make use of the fact that the isomorphism (2.7) generates a subdivision of B :

(2.8) $\mathcal{B} = \{B_i\}_{i=1, \dots, n}$, where $B_i := P(T_i)$.

Analogously to the techniques in finite element methods, we can introduce basis functions ϕ_j on T which are adapted to the given triangle structure of \mathcal{T} . Via the map P , this finite element structure can easily be transported onto the surface B , in a manner conforming to the subdivision (2.8).

In order to generate the matrix entries (1.5), we typically would have to calculate integrals of the form

$$(2.9) \quad M[k, j] = \sum_{i \in I(j)} \int_{B_i} \frac{\partial s(x_k, y)}{\partial \nu(y)} \phi_j(y) \mu(dy),$$

where j and k are the running indices of the matrix, i is the index of the subdivision (2.8), and the index set $I(j)$ in the summation is typically very small, describing a few neighboring pieces of \mathcal{B} supporting the basis function ϕ_j . This basis function is typically generated by a piecewise polynomial function $\tilde{\phi}_j$ on T , i.e.,

$$(2.10) \quad \phi_j(P(z)) = \tilde{\phi}_j(z) \text{ for } z \in T.$$

(2.11) For all $i \in I(j)$, the function $\tilde{\phi}_j(z)$, $z \in T_i$ is a polynomial in the barycentric coordinates with respect to the vertices of T_i .

(2.12) $\tilde{\phi}_j$ vanishes on T_i for $i \notin I(j)$.

We propose to numerically approximate the integrals of the form (2.9) via the transformation P . Thus, (2.9) equals

$$(2.13) \quad M[k, j] = \sum_{i \in I(j)} \int_{T_i} \frac{\partial s(x_k, P(z))}{\partial \nu(P(z))} \phi_j(P(z)) P_*\mu(dz).$$

Here, $P_*\mu$ indicates the measure on T which is obtained from the standard surface area measure μ on B via the map P , i.e., $P_*\mu(A) := \mu(P(A))$ for all measurable sets $A \subset T$.

Hence, our interest is concentrated on the numerical approximation of integrals

$$(2.14) \quad \int_{\Delta} f(P(z)) P_*\mu(dz),$$

where Δ is a triangle of a pseudomanifold approximating \mathcal{T} of B , and the integrand $f(P(z))$ can be numerically approximated on Δ and occasionally may have a weak singularity at a point of Δ .

In principle, we could use two directional derivatives $\partial_1 P$ and $\partial_2 P$ of P with respect to an orthogonal coordinate system of the Δ -plane to obtain

$$(2.15) \quad P_*\mu(dz) = \|\partial_1 P(z) \times \partial_2 P(z)\| \lambda(dz),$$

where λ is the standard Lebesgue measure on the plane. Hence, the problem is reduced to the numerical approximation of a standard multiple integral, and there are many methods available, see, e.g., Davis and Rabinowitz [6] or Stroud [18]. Efficient methods have also been discussed for the case of spherical coordinates (see Atkinson [5] or Keast [14]).

However, in this paper we consider the case where the map P is only given via a numerical algorithm (except in very simple cases), and hence we would have to use numerical approximations of the directional derivatives. Instead, we prefer to introduce a new approach, namely, a numerical method for directly approximating the integral (2.13), i.e., without making use of the directional derivatives of P .

3. The integration method. We propose to approximate the integral (2.14) by a repeated use of the trapezoidal rule

$$(3.1) \quad \int_{\Delta} f(P(z)) P_*\mu(dz) \approx \text{mean}_{i=1}^3 \left(f(P(v_i)) \right) \text{area}_{i=1}^3(P(v_i)),$$

where v_1, v_2, v_3 are the vertices of the triangle Δ , $\text{mean}_{i=1}^3(a_i)$ indicates the mean value of the numbers a_1, a_2, a_3 , and $\text{area}_{i=1}^3(w_i)$ indicates the area of a triangle composed of the vertices w_1, w_2, w_3 . Note that this rule differs from the standard trapezoidal rule

$$(3.2) \quad \int_{\Delta} f(P(z)) P_*\mu(dz) \approx \text{mean}_{i=1}^3 \left(f(P(v_i)) \gamma(v_i) \right) \text{area}_{i=1}^3(v_i),$$

which is obtained from multiple integration methods via the formula (2.15) by setting

$$\gamma(v) = \|\partial_1 P(v) \times \partial_2 P(v)\|.$$

We use the trapezoidal rule (3.1) in an adaptive refinement procedure as indicated in Fig. 1 below:

Fig. 1. *Refining the triangle.*

First we calculate a coarse approximation int_1 via (3.1), then we subdivide the triangle $[v_1, v_2, v_3]$ into four triangles

$$[v_1, v_{12}, v_{31}], [v_{12}, v_2, v_{23}], [v_{12}, v_{23}, v_{31}], [v_{31}, v_{23}, v_3],$$

and the trapezoidal rule (3.1) is applied to all four triangles. By summing up these values, we obtain a refined approximation int_2 of the given integral. The two approximations are compared. If they differ by less than a given tolerance “`TolInt`” (see the tolerance test below), then the algorithm stops. Otherwise, each of the four triangles is treated similarly. A recursive call of this rule enables the algorithm to locally adjust the refinement according to the given tolerance, as indicated in Fig. 1.

To be more precise, we give below a verbatim listing of the essential part of the algorithm, namely, the recursively written PASCAL procedure “`trapez.`” The procedure “`newton`” calculates the retraction map P .

```

procedure trapez(var v1, v2, v3, p1, p2, p3: vector;
                 var f1, f2, f3: real; level: integer);
var
  int1, int2: real;           {approximate integral values}
  v12, v23, v31, p12, p23, p31: vector;       {refinement}
  f12, f23, f31: real;       {integrand values}
  i: integer;

begin
if (level < leveldim)
then
  begin
  writeln('level exceeded');
  writeln(myfile,'level exceeded');
  close(myfile);
  halt;
  end;
for i:= 1 to 3 do
  begin
  v12[i]:= 0.5 * (v1[i] + v2[i]);
  v23[i]:= 0.5 * (v2[i] + v3[i]);
  v31[i]:= 0.5 * (v3[i] + v1[i]);
  end;
newton(v12, p12); newton(v23, p23); newton(v31, p31);
f12:= integrand(p12);
f23:= integrand(p23);
f31:= integrand(p31);
int1:= area(p1, p2, p3 ) * mean(f1, f2, f3 );
int2:= area(p1, p12, p31) * mean(f1, f12, f31)
      + area(p12, p2, p23) * mean(f12, f2, f23)
      + area(p12, p23, p31) * mean(f12, f23, f31)
      + area(p31, p23, p3 ) * mean(f31, f23, f3 );
if (abs(int1-int2) < TolInt)           {tolerance test}
then
  begin
  integral:= integral + int2;
  levstat[level]:= levstat[level] + 1;   {for statistical purposes}
  end
else
  begin
  trapez(v1, v12, v31, p1, p12, p31, f1, f12, f31, level+1);
  trapez(v12, v2, v23, p12, p2, p23, f12, f2, f23, level+1);
  trapez(v12, v23, v31, p12, p23, p31, f12, f23, f31, level+1);
  trapez(v31, v23, v3, p31, p23, p3, f31, f23, f3, level+1);
  end;
end;

```

4. Examples. In order to test the integration method, we use a simple example, namely, $H(x) := \|x\|^2 - 1$, so that $B = \{x \in \mathbf{R}^3 : H(x) = 0\}$ is the unit sphere. The vertices of the triangle $\Delta := [e_1, e_2, e_3]$ are the standard unit basis vectors. This gives a big triangle; usually the triangles of a triangulation will be much smaller and closer to the surface. Hence this numerical example displays many more refinements than will be usual. The projection $P(\Delta)$ is a spherical triangle which cuts out $\frac{1}{8}$ of the sphere. We consider two integrals. The first just describes the area of $P(\Delta)$:

$$(4.1) \quad I_1 := \int_{P(\Delta)} 1 \mu(dy) = \int_{\Delta} 1 P_*\mu(dz) = \frac{\pi}{2}.$$

The second has a weak singularity in $x = e_1$ at one of the vertices of the triangle. By interpreting the integral as a solid angle, see, e.g., [11] or [16], an exact value is readily obtained:

$$(4.2) \quad I_2 := \int_{P(\Delta)} \frac{\partial s(x, y)}{\partial \nu(y)} \mu(dy) = \int_{\Delta} \frac{\partial s(x, P(z))}{\partial \nu(P(z))} P_*\mu(dz) = \frac{\pi}{4}.$$

Below are two tables indicating the performance of the integration method for varying tolerances “**TolInt**.” It is important to note that the calculations were performed with a relative machine precision of $\approx 10^{-7}$.

Table 4.1

*The first table refers to the area (4.1) of the spherical triangle. As we expected, the refining of the various triangles is very uniform since no singularities occur. The global error is significantly bigger than the local errors which are \leq **TolInt** since the latter are accumulated. We show statistics of the number of triangles which were accepted via the tolerance test at each level of refinement.*

TolInt =	0.01	0.001	0.0001	0.00001	0.000001
Num. Int. I =	1.5510229	1.5658103	1.5695473	1.5703585	1.5705895
error: $\frac{\pi}{2} - I =$	0.0197734	0.004986	0.0012490	0.0004378	0.0002068
Level	Triangles	Triangles	Triangles	Triangles	Triangles
1	0	0	0	0	0
2	0	0	0	0	0
3	16	0	0	0	0
4	0	64	0	0	0
5	0	0	256	54	0
6	0	0	0	808	768
7	0	0	0	0	1024

In the above method, it is important to decide what has to be done about the singular values of the integrand f . Let us discuss the replacing of the singular value of f by a big number β , more precisely, we substitute for f the function

$$\tilde{f}(y) := \begin{cases} f(y) & \text{for } |f(y)| \leq \beta, \\ \beta & \text{for } f(y) > \beta, \\ -\beta & \text{for } f(y) < -\beta. \end{cases}$$

If the singular point x is a vertex point of the refinement, then it is easy to see that, due to the tolerance test, the method refines towards this point until approximately

$$\beta\alpha \approx \text{TolInt},$$

Table 4.2

The next table refers to the integral (4.2). Since the integrand has a weak singularity at the vertex $x = e_1$, we observe an iterated refining process towards this point. However, the overall numerical effort is not influenced very much by this local refinement since most triangles are accepted via the tolerance test at a reasonably low level of refinement.

TolInt =	0.01	0.001	0.0001	0.00001	0.000001
Num. Int. I =	0.77329779	0.78282577	0.78384632	0.78498906	0.78525865
error: $\frac{\pi}{4} - I =$	0.01210038	0.0025724	0.00155184	0.0004091	0.00013951
Level	Triangles	Triangles	Triangles	Triangles	Triangles
3	15	0	0	0	0
4	3	63	44	0	0
5	3	3	79	191	0
6	3	3	3	259	959
7	4	3	3	3	259
8	0	3	3	3	3
9	0	3	3	3	3
10	0	3	3	3	3
11	0	4	3	3	3
12	0	0	3	3	3
13	0	0	3	3	3
14	0	0	4	3	3
15	0	0	0	3	3
16	0	0	0	3	3
17	0	0	0	4	3
18	0	0	0	0	3
19	0	0	0	0	3
20	0	0	0	0	3
21	0	0	0	0	4

where α is the area of the triangle Δ of the refinement having vertex x . On the other hand, if the integrand f has a singularity of the form

$$f(y) \sim \frac{1}{\|x - y\|},$$

it is easy to see by using polar coordinates that

$$\int_{P(\Delta)} f(y) \mu(dy) = O(\delta),$$

where δ measures the diameter of Δ . Hence the method has been designed to neglect this integral piece if

$$\delta \approx \text{TolInt}.$$

By making the coarse approximation

$$\alpha \approx \delta^2,$$

it now becomes clear that it is sufficient to set

$$\beta \approx \frac{1}{\text{TolInt}}$$

in order to achieve the desired accuracy through the adaptive refinement. This technique was used in the numerical example (4.2). Numerical experiments confirm the

above discussion: if β is increased further, then only the number of refinements near the singularity increases without essentially improving the accuracy of the global integral (4.2).

5. Extrapolation. It is very suggestive to try an extrapolation method for the case that the integrand in (2.14) is sufficiently smooth on $P(\Delta)$. However, an asymptotic expansion of the composite trapezoidal rule as a theoretical background for an extrapolation method is lacking in our case. Below we conjecture such an expansion theorem which will be confirmed by the numerical experiments. Let the triangle Δ of a triangulation \mathcal{T} of B be given. For each integer $n > 0$, we consider an equidistant subdivision of Δ into n^2 triangles $\{\Delta_i\}_{i=1,\dots,n^2}$ by drawing $n - 1$ parallel lines to each edge of Δ such that each edge is subdivided into n equidistant intervals; see Fig. 2 below for the case $n = 5$.

Fig. 2. *Equidistant subdivision of a triangle.*

We now subdivide the integral (2.14) correspondingly:

$$(5.1) \quad \int_{\Delta} f(P(z)) P_*\mu(dz) = \sum_{i=1}^{n^2} \int_{\Delta_i} f(P(z)) P_*\mu(dz)$$

and approximate each summand via the trapezoidal rule (3.1). Thus we obtain a composite trapezoidal rule for approximating (5.1), which we call $I(n)$. Now our conjecture can be formulated.

CONJECTURE 5.1. *Let $k > 0$ be an integer. Under the assumption that the map H and the integrand f are sufficiently smooth, there exist constants $\{c_\kappa\}_{\kappa=1,\dots,k} \subset \mathbf{R}$ which are independent of the degree of subdivision n such that*

$$\int_{\Delta} f(P(z)) P_*\mu(dz) - I(n) = \sum_{\kappa=1}^k \frac{c_\kappa}{n^{2\kappa}} + O\left(\frac{1}{n^{2k+2}}\right).$$

For the approximation of one-dimensional integrals

$$\int_0^1 g(t) dt$$

via the standard composite trapezoidal rule, such asymptotic expansion theorems are obtained from the famous Euler–Maclaurin sum formula, see, e.g., [17, §3.3], and

similar formulas were obtained for multiple integrals in [15]; see also [18, §3.17]. Unfortunately, because of the difference between our trapezoidal rule (3.1) and the standard trapezoidal rule (3.2), this does not imply the Conjecture 5.1. This subject will be pursued elsewhere.

Example 5.2. We consider again the numerical example (4.1), which we now approximate via our composite trapezoidal rule as described following (5.1). The table below gives the numerical results for increasing degree of subdivision n . The table is given in the typical fashion of the Romberg integration, see, e.g., [17, p. 133]. We note again that the relative machine error of the calculation was $\approx 10^{-7}$.

(5.2)	n $I(n)$
	4 1.4943614
	8 1.5510229 1.5699101
	16 1.5658103 1.5707395 1.5707948

For the corresponding errors, we obtain the table

(5.3)	n $\frac{\pi}{2} - I(n)$
	4 0.0764349
	8 0.0197734 0.0008863
	16 0.0049860 0.0000569 0.0000016

This is strong numerical evidence for the validity of Conjecture 5.1.

6. Concluding remarks. The procedure in §3 was written in a concise way in order to make the logical structure easier to understand. However, this was done at the cost of computational efficiency since for many points the retraction “`newton`” and the function evaluation “`integrand`” are calculated twice. To avoid this, a much more complex program must be written. The present paper is merely intended to underline the usefulness of our approach, which differs from standard techniques in integration theory. For planar coordinates, sophisticated and efficient adaptive integration routines have been developed; see, e.g., Friedman and Wright [7] and Kahaner and Rechar [13]. It would be desirable to adapt these or similar techniques to the case which is considered here.

The procedure in §3 is not suitable for use with an extrapolation method. A modification has to be written for this important case. Furthermore, since the computational efficiency of this method mainly depends on the number of triangles which are generated, the Bulirsch numbers $n = 1, 2, 3, 4, 6, 8, 12, \dots$ should be used; see [17, eq. (3.4.5)(b)]. It is also pointed out there that rational interpolation is more efficient than polynomial interpolation in this context.

For the typical application (2.9) that we have in mind, it is known a priori whether the integrand is singular or not. Note that singularity may also be assumed for numerical purposes if x_k is very near to but not in the piece B_i . It is therefore advisable to make an a priori choice of the integration method: an adaptive refinement procedure as indicated in §3 for the case in which the integrand is singular; and an extrapolation method as indicated in Example 5.2 for the case in which the integrand is sufficiently smooth. Since the triangles T_i are usually tangent to B and very small, very few refinement steps should provide for sufficient accuracy in both cases. Note that our numerical examples (4.1)–(4.2) have a very large triangle. This was chosen on purpose in order to better display the performance of the integration methods.

Acknowledgments. I strongly appreciated some enlightening discussions which I had with Gene Allgower from Colorado State University and with Ron Guenther from Oregon State University.

REFERENCES

- [1] E. L. Allgower and K. Georg, *Estimates for piecewise linear approximations of implicitly defined manifolds*, Appl. Math. Lett., 1 (1989), pp. 1–7.
- [2] ———, *Numerical Continuation Methods: An Introduction*, Springer-Verlag, Berlin, Heidelberg, New York, 1990.
- [3] 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.
- [4] K. E. Atkinson, *The numerical solution of Laplace's equation in three dimensions*, SIAM J. Numer. Anal., 19 (1982), pp. 263–274.
- [5] ———, *Numerical integration of the sphere*, J. Austral. Math. Soc. Ser. B, 23 (1982), pp. 332–347.
- [6] P. J. Davis and P. Rabinowitz, *Methods of Numerical Integration*, Second Edition, Academic Press, Orlando, FL, 1984.
- [7] J. H. Friedman and M. H. Wright, *A nested partitioning procedure for numerical multiple integration*, ACM Trans. Math. Software, 7 (1981), pp. 76–92.
- [8] S. Gnutzmann, *Stückweise lineare Approximation implizit definierter Mannigfaltigkeiten*, Ph.D. thesis, Department of Mathematics, University of Hamburg, Federal Republic of Germany, 1989.
- [9] R. B. Guenther and J. Lee, *Partial Differential Equations of Mathematical Physics and Integral Equations*, Prentice-Hall, Englewood Cliffs, NJ, 1987.
- [10] N. M. Günter, *Die Potentialtheorie und ihre Anwendung auf Grundaufgaben der Mathematischen Physik*, B. G. Teubner, Leipzig, German Democratic Republic, 1957.
- [11] W. Hackbusch, *Integralgleichungen. Theorie und Numerik*, B. G. Teubner, Stuttgart, Federal Republic of Germany, 1989.
- [12] M. A. Jaswon and G. T. Symm, *Integral Equation Methods in Potential Theory and Electrostatics*, Academic Press, New York, 1977.
- [13] D. K. Kahaner and O. W. Rechar, *TWODQD an adaptive routine for two-dimensional integration*, J. Comput. Appl. Math., 17 (1987), pp. 215–234.
- [14] P. Keast, *Cubature formulas for the surface of the sphere*, J. Comput. Appl. Math., 17 (1987), pp. 151–172.
- [15] J. N. Lyness and B. J. J. McHugh, *Integration over multidimensional hypercubes. 1: A progressive procedure*, Comput. J., 6 (1963), pp. 264–270.
- [16] S. G. Mikhlin, *Linear Integral Equations*, Hindustan Publishing Corp., Delhi, India, 1960.
- [17] J. Stoer and R. Bulirsch, *Introduction to Numerical Analysis*, Springer-Verlag, Berlin, Heidelberg, New York, 1980.
- [18] A. H. Stroud, *Approximate Calculation of Multiple Integrals*, Prentice-Hall, Englewood Cliffs, NJ, 1971.