Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 226 (2007) 1432–1443
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

Geometric integration over irregular domains with application to level-set methods

Chohong Min [a], Frédéric Gibou [b,*]

[a] Mathematics Department, KyungHee University, Seoul, Republic of Korea
[b] Mechanical Engineering Department & Computer Science Department, University of California, Santa Barbara, CA 93106, USA

Received 7 October 2006; received in revised form 27 May 2007; accepted 30 May 2007
Available online 14 June 2007

Abstract

We present a geometric approach for calculating integrals over irregular domains described by a level-set function. This procedure can be used to evaluate integrals over a lower dimensional interface and may be used to evaluate the contribution of singular source terms. This approach produces results that are second-order accurate and robust to the perturbation of the interface location on the grid. Moreover, since we use a cell-wise approach, this procedure can be easily extended to quadtree and octree grids. We demonstrate the second-order accuracy and the robustness of the method in two and three spatial dimensions.

Keywords: Integration; Level-set methods; Quadtree/octree data structures; Isosurfacing

1. Introduction

Integration over interfaces and irregular domains defined by a level-set function ϕ is traditionally computed using regularized one-dimensional Dirac delta or Heaviside functions [12,11]: Consider a domain $\Omega \subset R^n$ and a lower dimensional interface Γ separating two disjoints subdomains Ω^- and Ω^+. A level-set function ϕ can be used to represent Γ as the set of points x such that $\phi(x) = 0$, Ω^- by $\phi(x) \leqslant 0$ and Ω^+ by $\phi(x) > 0$. The integral of a function f on Γ and the integral of f over the subdomain Ω^- can be calculated as:

$$\int_\Gamma f \, \mathrm{d}\Gamma = \int_{\mathbb{R}^n} f(x) \cdot \delta(\phi(x)) \cdot \|\nabla \phi\| \, \mathrm{d}x, \tag{1}$$

$$\int_{\Omega^-} f \, \mathrm{d}\Omega = \int_{\mathbb{R}^n} f(x) \cdot [1 - H(\phi(x))] \, \mathrm{d}x, \tag{2}$$

where δ and H are approximations of the delta and Heaviside functions, respectively.

* Corresponding author. Tel.: +1 8058937152.
E-mail address: fgibou@engineering.ucsb.edu (F. Gibou).

0021-9991/$ - see front matter
doi:10.1016/j.jcp.2007.05.032

However, rather recently, Engquist et al. [2] pointed out that the above approach using standard delta formulations may lead to non-convergent approximations. The authors then presented a discretization of the Dirac delta function that removes the problem of convergence. Later, Smereka [17] proposed first- and second-order accurate discretizations of the regularized delta function using the work of Mayo [7]. Regularized delta functions serve not only in the evaluation of integrals over a lower dimensional surface, but also in the direct discretization of singular source terms, for example the surface tension force between two materials. The work of Engquist et al. [2] and Smereka [17] can therefore be used to directly evaluate singular source terms. However, singular forces can also be computed using integration over lower dimensional surfaces. In the case of the two-phase flow example, the force due to surface tension only exists at the interface between the two phases. In this case, the computation of the singular source term at a grid node adjacent to the interface can be approximated by the integral over the interface in the control volume element centered at the grid node. Computing integrals over lower dimensional surfaces is therefore an important task for evaluating integrals or for evaluating the contribution of singular source terms.

In this paper, we present a geometric approach for calculating integrals such as (1) and (2). This approach produces results that are second-order accurate and independent of the interface location on the grid. We discretize the interface into a disjoint union of simplices [1,8] and then use a numerical integration quadrature rule on simplices [4]. This direct discretization of the interface using tools from computational geometry produces theoretically sound and numerically efficient algorithms. We also compare the results obtained with this method to those obtained with the delta formulation of [17] and find that a geometric approach is less sensitive to small perturbations of the interface location.

2. Numerical integration

In this section, we present a numerical integration over the interface $\Gamma = \{x \in \mathbb{R}^n | \phi(x) = 0\}$ and the irregular domain $\Omega^- = \{x \in \mathbb{R}^n | \phi(x) \leqslant 0\}$ of a continuous function $\phi : \mathbb{R}^n \to \mathbb{R}$. Although this algorithm can be easily extended to higher dimensions, we focus on the two- and three-dimensional cases.

2.1. Triangulation of a grid cell

The approach we choose is to decompose grid cells crossed by the interface into a union of simplices, for which integration is straightforward. A simplex, or n-simplex is the convex hull of a set of $(n+1)$ affinely independent points, i.e. a triangle in two spatial dimensions and a tetrahedron in three spatial dimensions. The affinely independent points are called the vertices of the simplex, and the dimension of the simplex S is n, and denoted by $\dim(S)$.

Simplices provides a straightforward way to compute lengths, areas and volumes. Consequently, there have been thorough studies to decompose geometric shapes into simplices for many applications, for example meshing in finite elements [3]. In two spatial dimensions a grid cell can be decomposed into two triangles, as shown in Fig. 1. In three spatial dimensions a grid cell can be decomposed into five tetrahedra (called the middle cut triangulation [14]) or into six tetrahedra (called the Kuhn triangulation [6]). The Kuhn triangulation can be more easily extended to higher dimensions [8] and is more convenient to match triangulations between

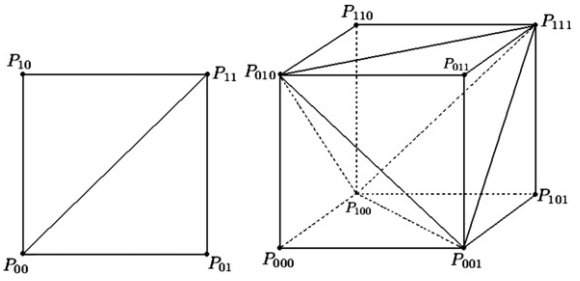

Fig. 1. Triangulation of a two- (left) and three- (right) dimensional cell.

adjacent cells. On the other hand, the angles of the tetrahedra created by the middle cut triangulation are less acute than those of the Kuhn triangulation and the smaller number of tetrahedra leads to faster computations. In this work, we choose the middle cut triangulation since we limit ourselves to two and three spatial dimensional cases and because integrations can be computed cell-wise, with no interactions between adjacent cells.

Cartesian cells can be mapped onto the unit cells, so with no loss of generality, we present the middle cut triangulation on the unit cells in two and three spatial dimensions as illustrated in Fig. 1:

$$[0,1]^2 = \mathrm{conv}(P_{00}, P_{10}, P_{11}) \cup \mathrm{conv}(P_{00}, P_{01}, P_{11})$$

and

$$\begin{aligned}[0,1]^3 = {} & \mathrm{conv}(P_{000}, P_{100}, P_{010}, P_{001}) \cup \mathrm{conv}(P_{110}, P_{100}, P_{010}, P_{111}) \cup \mathrm{conv}(P_{101}, P_{100}, P_{111}, P_{001}) \\ & \cup \mathrm{conv}(P_{011}, P_{111}, P_{010}, P_{001}) \cup \mathrm{conv}(P_{111}, P_{100}, P_{010}, P_{001}),\end{aligned}$$

where $\mathrm{conv}(P_1, \ldots, P_n)$ is the convex hull span by the points $P_1, \ldots, P_n$ and defines a simplex. We denote by $T(C)$ the triangulation of a cell C, i.e. C is a disjoint union of simplices in $T(C)$:

$$C = \bigcup_{S \in T(c)} S.$$

Note that the triangulations do not create new vertices, thus a discrete function sampled on a uniform grid is well defined on the vertices of each simplex. Then the integrations are calculated simplex-wise as:

$$\int_\Gamma f \, \mathrm{d}\Gamma = \sum_{C:\text{grid cell}} \int_{C\cap\Gamma} f \, \mathrm{d}\Gamma = \sum_{C:\text{grid cell}} \sum_{S\in T(C)} \int_{S\cap\Gamma} f \, \mathrm{d}\Gamma,$$

$$\int_{\Omega^-} f \, \mathrm{d}\Omega = \sum_{C:\text{grid cell}} \int_{C\cap\Omega^-} f \, \mathrm{d}\Omega = \sum_{C:\text{grid cell}} \sum_{S\in T(C)} \int_{S\cap\Omega^-} f \, \mathrm{d}\Omega.$$

The integrations are then reduced to $\int_{S\cap\Gamma} f \, \mathrm{d}\Gamma$ and $\int_{S\cap\Omega^-} f \, \mathrm{d}\Omega$ for each simplex S, which necessitate a discretization of $S \cap \Gamma$ and $S \cap \Omega^-$.

2.2. Discretization of $S \cap \Gamma$ and $S \cap \Omega^-$

In general, $S \cap \Gamma$ and $S \cap \Omega^-$ are continuous manifolds possibly with some singularities (sharp corners or sharp edges). For simpler calculations, we approximate the sets with the linear interpolation of ϕ, using the ϕ values on the vertices of the simplex S, as in [8]. Then, $S \cap \Gamma$ and $S \cap \Omega^-$ are defined by polytopes, i.e. convex hulls of finite points. This allows for efficient representation, since only a finite number of points need to be stored. Specifically, we first approximate the location of the set Γ on each edge of S: Let $\{P_0, \ldots, P_n\}$ be the vertices of S, if Γ crosses the edge P_iP_j, i.e. $\phi(P_i)\phi(P_j) < 0$, we define the intersection point P_{ij} between Γ and the edge as:

$$P_{ij} = P_j \frac{\phi(P_i)}{\phi(P_i) - \phi(P_j)} - P_i \frac{\phi(P_j)}{\phi(P_i) - \phi(P_j)}.$$

Then $S \cap \Gamma$ and $S \cap \Omega^-$ are polytopes whose vertices are given by:

$$S \cap \Gamma \approx \mathrm{conv}(\{P_{ij} | \phi(P_i)\phi(P_j) < 0\}),$$

and

$$S \cap \Omega^- \approx \mathrm{conv}(\{P_{ij} | \phi(P_i)\phi(P_j) < 0\} \cup \{P_i | \phi(P_i) < 0\}),$$

as illustrated in Figs. 2–4.

Integration over such discretizations of $S \cap \Gamma$ or $S \cap \Omega^-$ can still be complicated since the area or volume computations of polytopes are in general not straightforward. For this reason, if the sets $S \cap \{\phi = 0\}$ or $S \cap \{\phi \leqslant 0\}$ are not simplices, we further divide them into simplices as described next: Using the values of ϕ at the vertices of S, we linearly interpolate ϕ inside S, thus Γ and Ω^- are geometrically hyperplane and half-space, respectively. First note that two geometric configurations given by the intersection between a simplex S

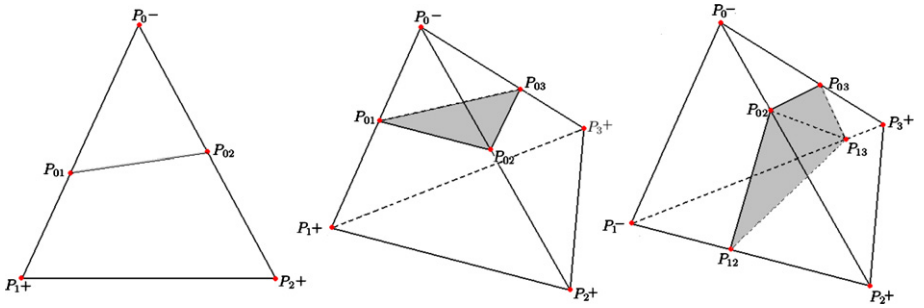

Fig. 2. Representation of the set $S \cap \Gamma$ when $\dim(S) = 2$ (left) and $\dim(S) = 3$ (center and right), which is approximated by $\mathrm{conv}(\{P_{ij} | \phi(P_i)\phi(P_j) < 0\})$, where the P_{ij}s are found by linear interpolations of ϕ. The left figure represents the generic case in two spatial dimensions and the right two figures represent the generic cases in three spatial dimensions.

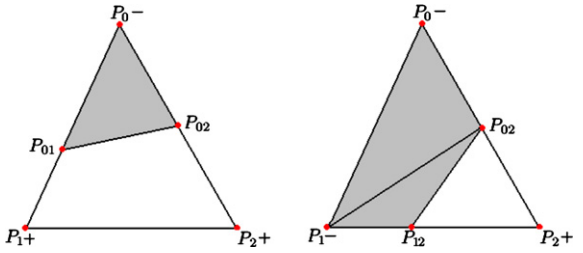

Fig. 3. The two generic representations of the set $S \cap \Omega^-$ in two spatial dimensions: One triangle (left) or the union of two triangles (right).

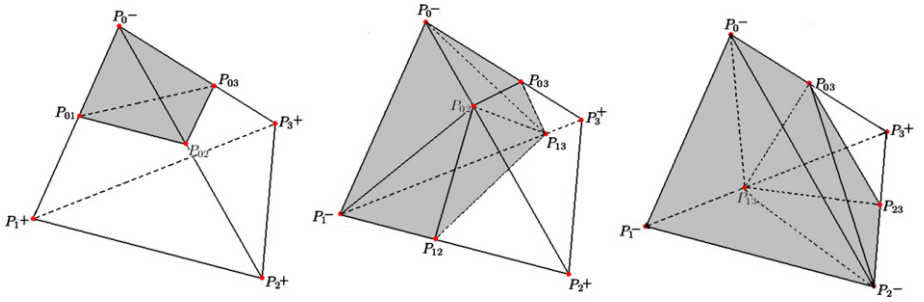

Fig. 4. The three generic representations of the set $S \cap \Omega^-$ in three spatial dimensions: One tetrahedron (left) or the union of three tetrahedra (center and right).

and a hyperplane Γ are equivalent if they have the same number of vertices below and above the hyperplane Γ, as noted in [18]. Likewise, two geometric configurations given by the intersection between a simplex S and a halfspace Ω^- are equivalent if they have the same number of vertices below and above the hyperplane Γ. Obviously, since the sum of the number of vertices below the hyperplane and the number of vertices above the hyperplane equals the total number of vertices of S, it is enough to know the number of vertices below the hyperplane to classify all the possible geometric configurations. We define this number by

$$\eta(\phi, S) := n(\{P_i | \phi(P_i) < 0\}).$$

In two spatial dimensions, the possible values of $\eta(\phi, S)$ are 0, 1, 2, and 3. When $\eta(\phi, S) = 0$ or 3, ϕ is positive or negative, therefore $S \cap \Gamma = \emptyset$. The case of $\eta(\phi, S) = 2$ can be treated in the same fashion as the case $\eta(\phi, S) = 1$ by simply negating ϕ (since $S \cap \{\phi = 0\} = S \cap \{-\phi = 0\}$). When $\eta(\phi, S) = 1$, $S \cap \Gamma$ is a line segment as illustrated in Fig. 2 (left) and Table 1 summarizes the different cases. We characterize the decomposition of $S \cap \Omega^-$ in the same way: When $\eta(\phi, S) = 0$, ϕ is positive, therefore $S \cap \Omega^- = \emptyset$. When $\eta(\phi, S) = 3$, ϕ is negative and $S \cap \Omega^- = S$. Fig. 3 illustrates the decompositions when $\eta(\phi, S) = 1$ (left) and $\eta(\phi, S) = 2$ (right). Table 3 summarizes the different cases.

Table 1
Generic case for representing $S \cap \Gamma$ in two spatial dimensions

$\phi(P_0)$	$\phi(P_1)$	$\phi(P_2)$	Q_0	Q_1
$-$	$+$	$+$	P_{01}	P_{02}

Q_0 and Q_1 are the vertices of a line segment as depicted in Fig. 2 (left).

Table 2
The two generic cases for representing $S \cap \Gamma$ in three spatial dimensions

$\phi(P_0)$	$\phi(P_1)$	$\phi(P_2)$	$\phi(P_3)$	Q_0	Q_1	Q_2
$-$	$+$	$+$	$+$	P_{01}	P_{02}	P_{03}
$-$	$-$	$+$	$+$	P_{02}	P_{03}	P_{13}
				P_{02}	P_{12}	P_{13}

Q_0, Q_1, and Q_2 are the vertices of a triangle as depicted in Fig. 2 (center and right).

Table 3
The two generic cases for representing $S \cap \Omega^-$ in two spatial dimensions

$\phi(P_0)$	$\phi(P_1)$	$\phi(P_2)$	Q_0	Q_1	Q_2
$-$	$+$	$+$	P_0	P_{01}	P_{02}
$-$	$-$	$+$	P_0	P_1	P_{02}
			P_1	P_{12}	P_{02}

Q_0, Q_1, and Q_2 are the vertices of a triangle as depicted in Fig. 3.

In three spatial dimensions, the possible values of $\eta(\phi, S)$ are 0, 1, 2, 3, and 4. When $\eta(\phi, S) = 0$ (resp. $\eta(\phi, S) = 4$), ϕ is positive (resp. negative) and $S \cap \Gamma = \emptyset$. The case $\eta(\phi, S) = 3$ can be treated in the same fashion as the case $\eta(\phi, S) = 1$ by negating ϕ. When $\eta(\phi, S) = 1$ and $\eta(\phi, S) = 2$, the decomposition of $S \cap \Gamma$ is illustrated in Fig. 2 (center and right) and Table 2 summarizes the different cases. Likewise, we characterize the decomposition of $S \cap \Omega^-$: When $\eta(\phi, S) = 0$, ϕ is positive, therefore $S \cap \Omega^- = \emptyset$. When $\eta(\phi, S) = 4$, ϕ is negative, therefore $S \cap \Omega^- = S$. Fig. 4 illustrates the decompositions when $\eta(\phi, S) = 1$ (left), $\eta(\phi, S) = 2$ (center) and $\eta(\phi, S) = 3$ (right), and Table 4 summarizes the different cases.

We now denote the decompositions of $S \cap \Gamma$ and $S \cap \{\phi \leqslant 0\}$ by $\mathrm{Sec}(\phi, S)$ and $\mathrm{Fru}(\phi, S)$, respectively:

$$S \cap \Gamma = \bigcup_{S' \in \mathrm{Sec}(\phi, S)} S',$$

$$S \cap \Omega^- = \bigcup_{S' \in \mathrm{Fru}(\phi, S)} S'.$$

Table 4
The three generic cases for representing $S \cap \Omega^-$ in three spatial dimensions

$\phi(P_0)$	$\phi(P_1)$	$\phi(P_2)$	$\phi(P_3)$	Q_0	Q_1	Q_2	Q_3
$-$	$+$	$+$	$+$	P_0	P_{01}	P_{02}	P_{03}
$-$	$-$	$+$	$+$	P_0	P_1	P_{02}	P_{13}
				P_{12}	P_1	P_{02}	P_{13}
				P_0	P_{03}	P_{02}	P_{13}
$-$	$-$	$-$	$+$	P_0	P_1	P_2	P_{13}
				P_0	P_{03}	P_2	P_{13}
				P_{23}	P_{03}	P_2	P_{13}

Q_0, Q_1, Q_2, and Q_3 are the vertices of a tetrahedron, as depicted in Fig. 4.

Using these decompositions, the integrals over Γ and $\{\phi \leqslant 0\}$ are defined as the sum of integrals over simplices:

$$\int_\Gamma f \, \mathrm{d}\Gamma = \sum_{C:\text{grid cell}} \sum_{S \in T(C)} \int_{S \cap \Gamma} f \, \mathrm{d}\Gamma = \sum_{C:\text{grid cell}} \sum_{S \in T(C)} \sum_{S' \in \mathrm{Sec}(\phi,S)} \int_{S'} f \, \mathrm{d}\Gamma,$$

$$\int_{\Omega^-} f \, \mathrm{d}\Omega = \sum_{C:\text{grid cell}} \sum_{S \in T(C)} \int_{S \cap \Omega^-} f \, \mathrm{d}\Omega = \sum_{C:\text{grid cell}} \sum_{S \in T(C)} \sum_{S' \in \mathrm{Fru}(\phi,S)} \int_{S'} f \, \mathrm{d}\Omega.$$

2.3. Numerical integration on a simplex

The integral of a function f over a simplex S with vertices $P_0, \ldots, P_n$ can be approximated by the following second-order midpoint rule [4]:

$$\int_S f \, \mathrm{d}x = \mathrm{vol}(S) \cdot \frac{f(P_0) + \cdots + f(P_n)}{n+1}.$$

We note that there exist higher order accurate quadrature rules to approximate integrals [4,5], but in our work the use of linear interpolation for ϕ limits the accuracy to second-order and therefore approximating the integral by the second-order midpoint rule is sufficient. The volume of the simplex is given by

$$\mathrm{vol}(S) = \frac{1}{n!} \left| \det \begin{pmatrix} (P_1 - P_0)e_1 & \cdots & (P_n - P_0)e_1 \\ \vdots & & \vdots \\ (P_1 - P_0)e_n & \cdots & (P_n - P_0)e_n \end{pmatrix} \right|,$$

where e_i represents the ith canonical unit basis vector.

Remarks

- In the case where $\phi = 0$ on an edge, the summation over the adjacent grid cell will duplicate the integral on the edge. Therefore, we exclude this case by simply perturbing ϕ. In the example section we use $\epsilon = 10^{-20}$, and let

$$\phi_{ijk} = \begin{cases} \epsilon & \text{if } |\phi_{ijk}| < \epsilon \text{ and } \phi_{ijk} > 0 \\ -\epsilon & \text{if } |\phi_{ijk}| < \epsilon \text{ and } \phi_{ijk} \leqslant 0. \end{cases}$$

- In the case where all the vertices of a cell C belongs to Ω^- in the calculation of $\int_{C \cap \Omega^-} f(x) \, \mathrm{d}\Omega$, the triangulation procedure is not necessary. By the standard second-order mid-point rule, the integration is approximated as the product of the volume of the cell and the average of f at the vertices of the cell.

3. Extension to quadtree and octree data structures

This approach can be trivially extended to unstructured Cartesian meshes since the algorithm is based on integrating over a grid cell, with no dependence from one cell to others. For the sake of efficiency, it is desirable to utilize data structures that reduce the total amount of cells to be used, since only those adjacent to the set Γ contribute to the integration of $\int_\Gamma f \, \mathrm{d}\Gamma$. In this section, we describe a simple implementation using quadtree and octree data structures.

3.1. Quadtree/octree data structures

Quadtree (resp. octree) data structures can be used to represent the spatial discretization of a physical domain in two (resp. three) spatial dimensions as depicted in Fig. 5: Initially the root of the tree is associated

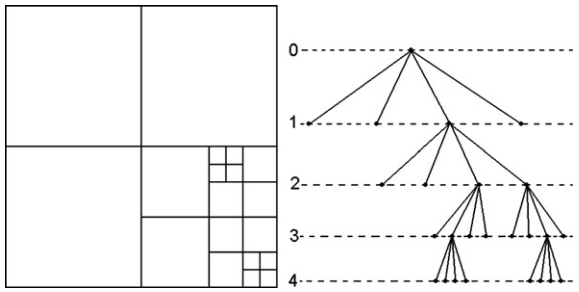

Fig. 5. Discretization of a two-dimensional domain (left) and its quadtree representation (right). The entire domain corresponds to the root of the tree (level 0). Then each cell can be recursively subdivided further into four children. In this example, the tree is ungraded since the difference of level between cells exceeds one.

with the entire domain, then we recursively split each cell into four children until the desired level of detail is achieved. This is done similarly in three spatial dimensions, except that cells are split into eight children. We refer the reader to the books of Samet [16,15] for more details on quadtree/octree data structures.

3.2. Constructing quadtrees/octrees from implicit representation of the domains

Since the accuracy of the method depends on the size of the cells adjacent to the interface, we impose that the finest cells lie on the interface. This can be achieved by using the signed distance function to the interface along with the Whitney decomposition, as first proposed by Strain in [19]. Simply stated, one "*splits any cell whose edge length exceeds its distance to the interface*". For a general function $\phi : \mathbb{R}^n \to \mathbb{R}$ with Lipschitz constant $Lip(\phi)$, the Whitney decomposition was extended in Min [10]: Starting from a root cell, split any cell C if

$$\max_{v \in \text{vertices}(C)} |\phi(v)| \leqslant \frac{1}{2} \cdot \text{Lip}(\phi) \cdot \text{diag-size}(C),$$

where diag-size(C) refers to the length of the diagonal of the current cell C and v refers to a vertex (node) of the current cell. We note that a signed distance function can be obtained from a given level-set function ϕ by using the so-called reinitialization equation, introduced for uniform grids by Sussman et al. [20] (see also Russo and Smereka [13]) and extended to unstructure grids in Min and Gibou [9].

Remarks

- The complexity of this approach scales with the dimension of the interface, i.e. $O(n)$ in two spatial dimensions and $O(n^2)$ in three spatial dimensions. In addition, since the isosurfacing is table-based, the computational time to evaluate the integral is similar to that using the delta formulation of [17].
- The reinitialization of the level-set function is only used in the construction of the octree grids. The computation of the integral itself only requires a continuous level-set function.

4. Examples

In this section, we demonstrate that our approach produces robust, second-order accurate results in two and three spatial dimensions. All the examples were computed on a PC with 2.2 GHz CPU and 2 GB memory.

4.1. Robustness to grid perturbations in 2D

Consider an ellipse represented as the zero level set of $\phi(x,y) = \frac{x^2}{1.5^2} + \frac{y^2}{.75^2} - 1$. Its exact arc-length is given as $\simeq 7.266336165$ [17]. Table 5 shows that the proposed scheme is second-order accurate and produces results that are stable in the case where the interface is randomly shifted on the grid. The robustness is demonstrated by the fact that the ratios of the maximum error over the minimum error (over all trials) in Table 5 are close to

Table 5
Statistics of 50 trials for computing the arc-length of an ellipse for Example 4.1

Δx	Average	Order	SD	Min	Order	Max	Order	$\frac{\max}{\min}$
Geometric numerical integration								
.2	5.04E−3		2.15E−4	4.63E−3		5.49E−3		1.19
.1	1.26E−3	2.00	3.23E−5	1.17E−3	1.99	1.30E−3	2.08	1.11
.05	3.14E−4	2.00	6.61E−6	3.03E−4	1.95	3.26E−4	2.00	1.08
.025	7.84E−5	2.00	1.25E−6	7.50E−5	2.02	7.99E−5	2.03	1.07
.0125	1.96E−5	2.00	2.15E−7	1.90E−5	1.98	1.99E−5	2.01	1.04
.00625	4.90E−6	2.00	3.18E−8	4.83E−6	1.98	4.94E−6	2.01	1.02
First-order delta function approach [17]								
.2	8.96E−3		7.39E−3	1.28E−4		2.67E−2		208
.1	2.70E−3	1.73	2.96E−3	9.13E−5	1.31	1.07E−2	0.49	118
.05	9.55E−4	1.50	1.12E−3	4.19E−7	1.28	4.43E−3	7.77	10600
.025	3.21E−4	1.57	3.58E−4	7.32E−6	1.54	1.52E−3	-4.12	208
.0125	1.13E−4	1.51	1.22E−4	9.15E−6	1.53	5.28E−4	-0.32	57.7
.00625	3.94E−5	1.52	4.17E−5	2.01E−6	1.52	1.84E−4	2.19	91.7
Second-order delta function approach [17]								
.2	3.23E−3		2.74E−3	6.07E−4		1.30E−2		21.5
.1	5.74E−4	2.49	5.25E−4	2.93E−6	7.69	3.02E−3	2.10	1030
.05	1.13E−4	2.34	4.08E−5	2.55E−5	3.12	2.04E−4	3.88	8.01
.025	3.08E−5	1.87	8.31E−6	1.56E−5	0.70	4.72E−5	2.11	3.00
.0125	7.61E−6	2.01	1.51E−6	1.37E−6	3.50	1.21E−5	1.96	8.82
.00625	1.89E−6	2.01	1.82E−7	1.61E−6	0.23	2.21E−6	2.45	1.37

Top: Geometric approach. Middle and bottom: The first- and second-order accurate delta formulation approaches of [17].

Table 6
Convergence rate for computing the arc-length of an ellipse for Example 4.1 for one trial

Δx	Geometric integral	Order	First-order delta	Order	Second-order delta	Order
0.2	5.49E−3		3.94E−3		8.22E−4	
0.1	1.24E−3	2.13	9.03E−3	−1.19	1.15E−2	−3.80
0.05	3.03E−4	2.03	4.42E−3	1.02	4.99E−3	1.20
0.025	7.49E−5	2.01	1.52E−3	1.54	1.41E−3	1.82
0.0125	1.90E−5	1.97	5.28E−4	1.52	3.66E−4	1.94

Comparison of the geometric approach with the first- and second-order accurate delta formulation approaches of [17].

one. We compare our results with those obtained with the first-order delta formulation of [17]. In this case, the ratios of the maximum error over the minimum error can be large for some configurations (see Table 5), indicating a dependence on the interface location on the grid. We also compare our results to the second-order delta formulation of [17]. In this case the integration is much less sensitive to the interface location on the grid, as shown in Table 5. Table 6 shows the results of the three approaches over one trial.

We point out that the second-order accurate formulation of [17] reduces to the first-order formulation in the case where the interface is aligned with the Cartesian directions so that second-order accuracy is not

Table 7
Relative errors on computing the area of an ellipse for Example 4.1

Δx	Geometric integral	Order
.2	1.59E−2	
.1	3.76E−3	2.08
.05	9.46E−4	1.99
.025	2.25E−4	2.07
.0125	5.78E−5	1.97
.00625	1.46E−5	1.99

always possible. We also note that the geometric approach depends only on the continuity of the level function, while the first- and second-order delta formulations depends on the first- and second-order differentiability of the level function, respectively. In the case where the level function is not guaranteed to be smooth, the geometric approach may be advantageous.

Table 7 illustrates the second-order accuracy of our method for the computation of the area of an ellipse.

4.2. Robustness to grid perturbations in 3D

Consider an ellipsoid represented as the zero level set of $\phi(x,y,z) = \frac{x^2}{1.5^2} + \frac{y^2}{.75^2} + \frac{z^2}{.5^2} - 1$. Its exact surface area is given as $\simeq 9.901821$ (see [17]). Table 8 demonstrates that the proposed scheme is second-order accurate and compares our results to those obtained in [17]. In both cases the interface location is shifted randomly. Our method is robust as it is the case in two spatial dimensions. In the case of the delta formulation of [17], the first-order accurate approximation produces robust results that are close to second-order accurate and comparable to ours, while the second-order accurate approximation is less robust (see Table 8).

Table 9 demonstrate second-order accuracy of our approach for computing the volume of the ellipsoid.

4.3. Surface integral on a torus

Consider a torus T described by the zero level set of $\phi(x,y,z) = (\sqrt{x^2+y^2} - 2)^2 + z^2 - 1$. We compute the surface integral $\int_T x^2 \, dS = 22\pi^2$ on the torus. Fig. 6 (left) illustrate the geometry of the surface and Table 10 demonstrates the second-order accuracy of our method.

Table 8
Statistics of 50 trials for computing the surface area of an ellipsoid for Example 4.2

Δx	Average	Order	SD	Min	Order	Max	Order	$\frac{\max}{\min}$
Geometric numerical integration								
.2	3.17E−2		2.90E−4	3.12E−2		3.22E−2		1.03
.1	7.91E−3	1.98	1.02E−5	7.89E−3	1.98	7.94E−3	2.02	1.00
.05	1.98E−3	2.00	6.81E−7	1.98E−3	2.00	1.98E−3	2.00	1.00
.025	4.94E−4	2.00	1.13E−7	4.94E−4	2.00	4.95E−4	2.00	1.00
First-order delta function approach [17]								
.2	3.03E−2		7.12E−3	1.75E−2		4.73E−2		1.49
.1	7.77E−3	1.96	2.26E−3	3.95E−3	2.14	1.32E−3	1.84	1.51
.05	2.12E−3	1.87	7.36E−4	6.39E−4	2.62	4.48E−3	1.56	2.16
.025	5.20E−4	2.03	1.36E−4	3.41E−4	0.91	8.51E−4	2.39	1.54
Second-order delta function approach [17]								
.2	6.86E−2		7.60E−2	1.01E−2		4.46E−1		44.4
.1	1.33E−2	2.37	8.38E−3	9.47E−4	3.41	2.91E−2	3.93	3.08
.05	2.60E−3	2.35	2.11E−3	2.49E−4	1.92	1.06E−2	1.45	42.4
.025	8.00E−4	1.70	1.01E−3	4.40E−6	5.82	6.88E−3	0.62	1570

Top: Geometric approach. Middle and bottom: The first- and second-order accurate delta formulation approaches of [17].

Table 9
Relative errors on computing the volume of an ellipsoid for Example 4.2

Δx	Geometric integral	Order
.1	1.36E−2	
.05	3.40E−3	2.00
.025	8.50E−4	2.00
.0125	2.12E−4	2.00
.00625	5.31E−5	2.00

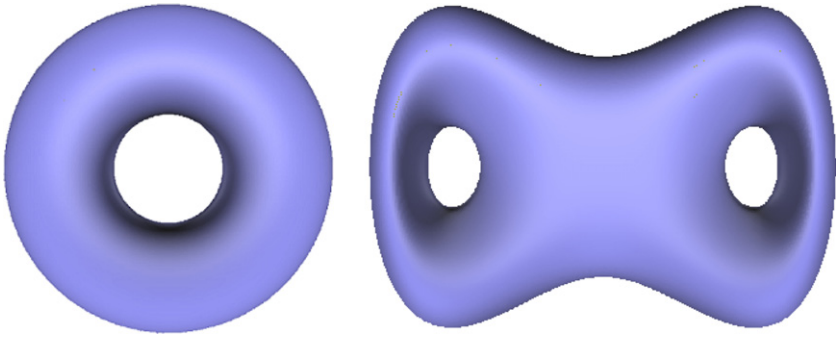

Fig. 6. Torus used in Example 4.3 (left) and genus two surface (right) used in Example 4.4.

4.4. Surface integral on a surface of genus two

Consider a surface represented by the zero level set of $\phi(x,y,z) = ((1.2 - x^2)x^2 - y^2)^2 + z^2 - .1$. According to the Gauss–Bonet theorem, the surface integral of the Gaussian curvature κ_G is -4π, since the surface has genus two, i.e. two handles as shown in Fig. 6 (right). The Gaussian curvature can be calculated from the level function as:

$$\kappa_G = \frac{\begin{array}{l}(\phi_x^2(\phi_{yy}\phi_{zz} - \phi_{yz}^2) + 2\phi_y\phi_z(\phi_{xy}\phi_{zx} - \phi_{yz}\phi_{xx}) \\ +\phi_y^2(\phi_{xx}\phi_{zz} - \phi_{xz}^2) + 2\phi_x\phi_z(\phi_{xy}\phi_{yz} - \phi_{xz}\phi_{yy}) \\ +\phi_z^2(\phi_{xx}\phi_{yy} - \phi_{xy}^2) + 2\phi_x\phi_y(\phi_{yz}\phi_{zx} - \phi_{xy}\phi_{zz}))\end{array}}{(\phi_x^2 + \phi_y^2 + \phi_z^2)^2}.$$

In this example, all the derivatives are approximated with the standard central finite differences. Table 11 demonstrates the second-order accuracy of our method.

4.5. Surface integral using octrees

Consider a surface, called orthocircles, represented as the zero contour of the level-set function:

$$\phi(x,y,z) = ((x^2 + y^2 - 1)^2 + z^2)((y^2 + z^2 - 1)^2 + x^2)((z^2 + x^2 - 1)^2 + y^2) - .075^2(1 + 3(x^2 + y^2 + z^2)).$$

Table 10
Relative errors on computing the surface integral for Example 4.3

Δx	Geometric integral	Order
.2	7.08E−3	
.1	1.78E−3	1.99
.05	4.49E−4	1.99
.025	1.12E−4	1.99
.0125	2.85E−5	2.00

Table 11
Relative errors on computing the surface integral for Example 4.4

Δx	Geometric integral	Order
.1	1.96E−1	
.05	6.38E−2	1.62
.025	1.65E−2	1.95
.0125	4.11E−3	2.01
.00625	1.03E−3	1.99

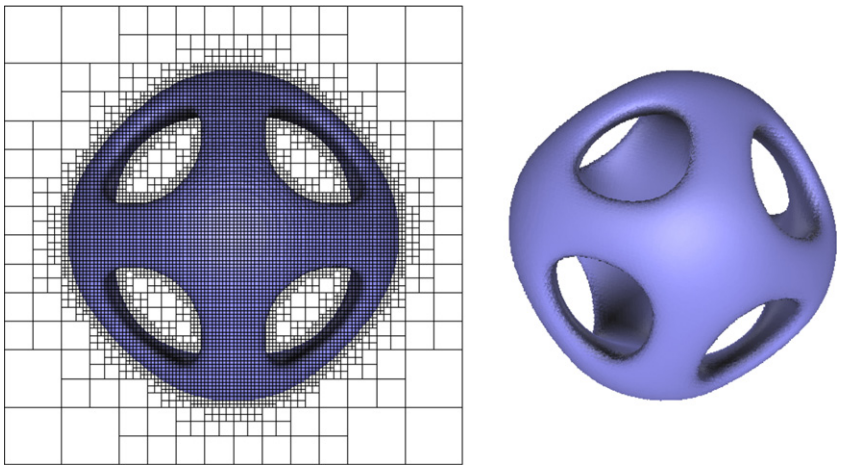

Fig. 7. Three-dimensional object (orthocircles) with genus 7 used in Example 4.5. Left: Octree representation of the interface demonstrating that the smallest cells are placed around the interface. Right: A different view of the same surface.

Table 12
Genus calculation for Example 4.5

Finest resolution	Δx	Number of nodes	Order	Relative error	Order	Genus	Order
64^3	.05	45,113		2.22-1		8.337614	
128^3	.025	206,241	2.19	4.95-2	2.17	7.297007	2.19
256^3	.0125	834,557	2.02	1.23-2	2.00	7.074171	2.01
512^3	.00625	3,372,753	2.01	3.08-3	2.00	7.018503	2.01
1024^3	.003125	13,546,163	2.00	7.69-4	2.00	7.004615	2.01

This surface is topologically equivalent to a surface with seven holes. We compute the surface integral of the Gaussian curvature as in the previous example but we use the octree data structure described in Section 3. Fig. 7 depicts the geometrical shape and the octree representation. From the surface integral of the Gaussian curvature, we derive the genus number g of the surface by the Gauss–Bonne theorem: $\int_\Gamma \kappa_G \, dx = 4\pi(1-g)$. Table 12 demonstrate the second-order accuracy of our method. Note that the number of nodes grows quadratically, since the octree data structure put most resource near the interface. The number of nodes for a 1024^3 uniform grid is about 80 times more than for the octree data structure used in this example, which produces the same effective resolution.

5. Conclusion

We have presented a geometric approach for calculating integrals over irregular domains described by a level-set function. This procedure can be used to evaluate integrals over a lower dimensional interface and may be used to evaluate the contribution of singular source terms. This approach produces results that are second-order accurate and robust to the perturbation of the interface location on the grid. Moreover, since we use a cell-wise approach, this procedure can be easily extended to quadtree and octree grids. We demonstrate the second-order accuracy and the robustness of the method in two and three spatial dimensions and compared with the first- and second-order accurate discretizations of the delta function formulation of [17].

Acknowledgments

The authors acknowledge stimulating discussions with Peter Smereka on the approximation of the delta function, and Guofang Wei for discussion on the genus in the orthocircles' example. The research of C. Min was supported in part by the Kyung Hee University Research Fund (KHU-20070608) in 2007. The research of F. Gibou was supported in part by a Sloan Research Fellowship in Mathematics.

References

[1] B.P. Carnerio, C. Silva, A.E. Kaufman, Tetra-cubes: an algorithm to generate 3d isosurfaces bases upon tetrahedra, Anais do IX SIBGRAPI, 1996, pp. 205–210.
[2] B. Engquist, A.K. Tornberg, R. Tsai, Discretization of dirac delta functions in level set methods, J. Comput. Phys. 207 (2005) 28–51.
[3] J.E. Goodman, J. O'Rourke, The Handbook of Discrete and Computational Geometry, CRC Press LL, 1997.
[4] A. Grundmann, M. Moeller, Invariant integration formulas for the *n*-simplex by combinatorial methods, SIAM J. Numer. Anal. 2 (1978) 282–290.
[5] C.P. Hammer, A.H. Stroud, Numerical integration over simplexes, Math. Tables Other Aids Comput. 10 (1956) 137–139.
[6] H.W. Kuhn, Some combinational lemmas in topology, IBM J. Res. Dev. 4 (1960) 508–524.
[7] A. Mayo, The fast solution of Poisson's and the biharmonic equations on irregular regions, SIAM J. Numer. Anal. 21 (1984) 285–299.
[8] C. Min, Simplicial isosurfacing in arbitrary dimension and codimension, J. Comput. Phys. 190 (2003) 295–310.
[9] C. Min, F. Gibou, A second order accurate level set method on non-graded adaptive Cartesian grids, Available at UCLA CAM report (06-22) http://www.math.ucla.edu/applied/cam/index.html, 2006 (in review).
[10] C.-H. Min, Local level set method in high dimension and codimension, J. Comput. Phys. 200 (2004) 368–382.
[11] S. Osher, J. Sethian, Fronts propagating with curvature-dependent speed: algorithms based on Hamilton–Jacobi formulations, J. Comput. Phys. 79 (1988) 12–49.
[12] D. Peng, B. Merriman, S. Osher, H. Zhao, M. Kang, A PDE-based fast local level set method, J. Comput. Phys. 155 (1999) 410–438.
[13] G. Russo, P. Smereka, A remark on computing distance functions, J. Comput. Phys. 163 (2000) 51–67.
[14] J.F. Sallee, The middle-cut triangulations of the *n*-cube, SIAM J. Alg. Disc. Methods 5 (1984) 407–419.
[15] H. Samet, The Design and Analysis of Spatial Data Structures, Addison-Wesley, New York, 1989.
[16] H. Samet, Applications of Spatial Data Structures: Computer Graphics, Image Processing and GIS, Addison-Wesley, New York, 1990.
[17] P. Smereka, The numerical approximation of a delta function with application to level set methods, J. Comput. Phys. 211 (2006) 77–90.
[18] D.M.Y. Sommerville, An Introduction to the Geometry of N Dimensions, Dover Publications, 1958.
[19] J. Strain, Tree methods for moving interfaces, J. Comput. Phys. 151 (1999) 616–648.
[20] M. Sussman, E. Fatemi, An efficient interface-preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow, SIAM J. Sci. Comp. 20 (1999) 1165–1191.