Part 6

Computer Science

Computational topology

Denis Blackmore and Thomas J. Peters

1. Introduction

The emphasis here will be upon how point-set topology can be applied to computing on geometric objects embedded in $\mathbb{R}^3$. The fundamental topological concept of a neighborhood generalizes limits over the reals, which inherently relies upon infinite precision arithmetic. Any specific computational representation of a real number is limited to being expressed in a finite number of bits. This cardinality disparity means that fundamental topological notions such as neighbhorhoods, dense sets and continuity are not well-expressed computationally, but can only be approximated. This presents novel opportunities for complementary research between topologists and numerical analysts.

The article *Computing over the reals: foundations for scientific computing* [**33**] begins,

> "The problems of scientific computing often arise from the study of continuous processes, and questions of computability and complexity over the reals are of central importance in laying the foundation for the subject."

The use of floating point numbers as an approximation of the reals entails a radically different perspective for classical point-set topologists, as the central topological notions regarding the interior, exterior and boundary of a set are based upon limits of infinite sequences of neighbhorhoods. These ideas are also crucial for geometric computations. Past practice can be somewhat tersely oversimplified as saying that the cardinality disparities have long been appreciated, but have been treated largely in an *ad hoc* fashion. Engineering practice and pragmatic programming, generally directed by heuristics, have been the dominant practice.

The definition adopted here for computational topology comes from the report *Emerging challenges in computational topology* [**21**]. (Also see Section 13.)

> "We intend the name *computational topology* to encompass both algorithmic questions in topology (for example, recognizing knots) and topological questions in algorithms (for example, whether a discrete construction preserves the topology of the underlying continuous domain)."

D. Blackmore gratefully acknowledges the partial support of this work from the NSF/DARPA CARGO grant CCR-0310619. T.J. Peters expresses appreciation for partial support from NSF grants DMS-0138098, CCR-022654, CCR-0429477, CCF-0533232 and by a 2005 IBM Faculty Award. All statements here are the responsibility of the authors, *not* of the National Science Foundation *nor* of the IBM Corporation.

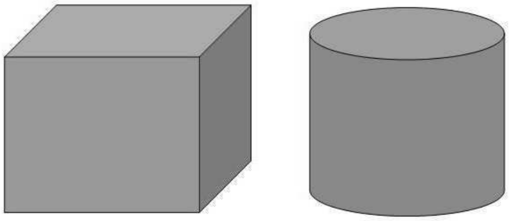


Figure 1. Box and cylinder

The broad definition is intended to prompt a "beneficial symbiosis" [**63**] between both subfields and "to extend computational geometry . . . into contact with classical topology" [**21**] with expected benefits to both fields. The subdiscipline of computational topology is relatively young. This very immaturity provides an important opportunity to consider its foundations as well to explore pernicious specific problems that remain unresolved.

2. History

The first usage of the term 'computational topology' appears to have occurred in the dissertation of M. Mäntylä [**120**]. The focus there was upon the connective topology joining vertices, edges and faces in geometric models, frequently also informally described as the *symbolic* information of a solid model. These vertices, edges and faces are discussed as the operands for the classical Euler operations.

2.1. Elementary manifold examples. In Figure 1, the box depicted on the left would have 8 vertices, 12 edges and 6 faces. This should be obvious, while the cylinder shown on the right entails an additional minor subtlety. Namely, the cylinder can be considered to be composed of an open cylinder and a top disc and a bottom disc. To include explicitly vertices and edges, the open cylinder will often be considered to be formed from a flat rectangle which has been rolled into a cylinder, with two opposing edges identified as one. This one edge would be vertical in the image on the right and would have a vertex at each end. Each disc would then be seen as having a circular bounding edge that had its initial and terminal vertex at one of these points on the vertical edge. This representation would then have 2 vertices, 3 edges and 3 faces (though other variants are clearly possible).

2.2. Non-manifold topology. Manifolds have a rich history in topology. They provide extensions of the usual topology on Cartesian products of the reals. Moreover, manifolds provide a generalization whereby points, curves, surfaces and solids have a common abstraction, but vary in dimension (from 0 to 3, respectively). Within the Boolean algebra of regular-closed, compact 3-manifolds, curves and surfaces are nowhere dense sets – meaning that the interior of their closures is empty within $\mathbb{R}^3$. Hence, these sets are trivial operands within that algebra. So, strict adherence to a programming paradigm, based upon this Boolean algebra of

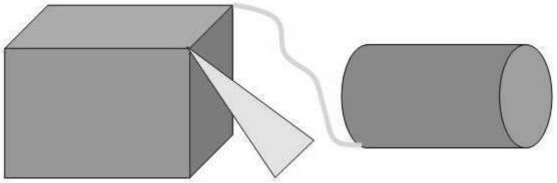


Figure 2. Model with non-manifold topology

3-manifolds would not admit directly the mixing of manifolds of differing dimension. Pioneering work by K.J. Weiler in his thesis [**176**] describing 'non-manifold topology' laid the intellectual framework for his initial prototype and extensive follow-up work by F. Printz in his 'Noodles' system [**88**].

Figure 2 shows how the manifolds of differing dimensions could be integrated in these systems to form one integrated geometric model. Each point need not have a neighborhood that is homeomorphic to a neighborhood in a 3-manifold (For 3-manifolds without boundary, these neighborhoods are just open neighborhoods of $\mathbb{R}^3$ and for 3-manifolds with boundary, the neighbhorhoods just have the usual relative topology associated with a boundary point). However, each point does have a neighborhood that is homeomorphic to an open neighborhood in an n-manifold, with n being equal to the lowest dimension of any of the manifolds joined at that point.

Question 2.1. *Is there a unifying topological abstraction covering manifolds, non-manifolds and other possible geometric models that might be useful to improve algorithmic design for geometric computations?* 1110?

Some other relevant references in the development of computational topology are listed [**13**, **50**–**52**, **57**, **59**, **137**–**139**, **172**].

3. Computation and the reals

Whenever computations are intended to be representative of operations on the reals, inherent concerns are the trade-offs required between algorithmic efficiency and sufficient numerical precision. This dilemma is discussed [**33**] relative to using a satisfactory number of terms from a Taylor's series approximation. The summarizing directive is "to take just enough terms *to satisfy our precision needs.*"

3.1. The role for tolerances. This same issue has been expressed within venues of the Society of Industrial and Applied Mathematics (SIAM) by the mathematician D.R. Ferguson and the engineer R. Farouki. Ferguson has observed that geometric models used in aeronautical and aerospace design require differing precisions depending upon the application software that is using such models for input [**75**]. His focus is upon the number of significant digits needed in order to have appropriate representations at topological boundaries formed from surface intersections. A broad overview of this concept is illustrated in the table in Figure 3. There the minimum precision needed for each application is given. Farouki

Application	Significant digits
visualization	2
computational fluid dynamics	3
multidisciplinary design optimization	4
computational electromagnetics	4
computational optics	7

FIGURE 3. Precision for differing applications

has espoused a similar point of view [**74**], based upon issues raised at a SIAM workshop that he and Ferguson organized with funding from the National Science Foundation (NSF).

This perspective raises several fundamental problems:

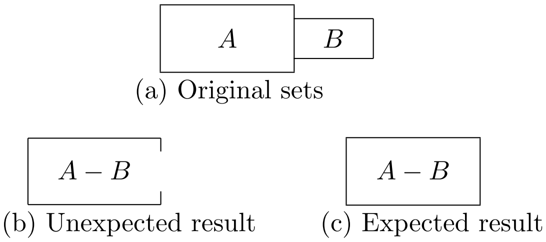
1111? **Question 3.1.** *What are the differing floating point precisions needed to capture accurately the topology along surface intersection boundaries in geometric models so that they can be reliably used in engineering simulations for visualization, computational fluid dynamics, stress analysis, computational optics, computational electromagnetics, etc.?*

1112? **Question 3.2.** *Are there crucially sensitive engineering applications that can be used to determine these precision needs? (For instance, are visualization and computational optics at extreme ends of the precision spectrum? Is understanding the needs of those two applications then sufficient for the conceptual framework for all modeling needs?)*

1113? **Question 3.3.** *Are some geometric intersection problems ill-conditioned?*

Question 3.4. *Will the process of finding the precision required for the models for these engineering simulations generalize to a mathematical methodology for being able to determine floating point precision needs for a wide variety of geometric models, inclusive of examples such as fractals and Julia sets?*

We note some abstractions that are appropriate for topologists. Individual computer science algorithms might be considered as specific functions, with distinctive domains and images. Even this level of abstraction is rarely articulated within computer science. Moreover, this view belies a cultural distinction between the computer science (CS) and mathematical communities. Topologists often focus attention of an entire family of functions, analyzing properties shared by an entire class of functions. For instance, homeomorphisms form an important such class within point-set topology, forming the basis for the traditional definition of topological equivalence. This broader approach, considering whole classes of algorithms, would be one way that topological perspectives can enrich CS. The process of going from one algorithm to another is then merely modeled by composition of functions. An example of how this view might also be useful in computer graphics is presented in Section 4.



(a) Original sets

(b) Unexpected result (c) Expected result

FIGURE 4. Subtraction of two sets

3.2. Engineering examples for computational topology. The material of this subsection is largely extracted from a related article [**137**] in order to introduce topologists to prominent engineering examples for computational topology.

The Boolean algebra of regular closed sets is prominent in topology, particularly as a dual for the Stone–Čech compactification. This algebra is also central for the theory of geometric computation, as a representation for combinatorial operations on geometric sets. However, the issue of computational approximation introduces unresolved subtleties that do not occur within *pure* topology.

The standard algorithmic operators on regular closed set representations are those from its Boolean algebra. These Boolean operations have an elegant symbolic representation in a binary tree, but do not typically include error bounds on the leaf node operands, which appears to fall within Knuth's definition [**104**] of algorithms being "properly called *seminumerical* because they lie on the borderline between numeric and symbolic calculation." This disparity between the theory and practice on this Boolean algebra is a central aspect of the "geometric robustness" problem [**96**].

The regular closed sets discussed here will be assumed to be subsets of $\mathbb{R}^3$, with its usual topology. The Boolean algebra of regular closed sets in $\mathbb{R}^3$ will be denoted as $\mathcal{R}(\mathbb{R}^3)$. Furthermore, any regular closed set considered will be assumed to be compact. Any surfaces and curves considered will be assumed to be compact 2-manifolds and 1-manifolds, respectively. All neighborhoods will be assumed to be open subsets of $\mathbb{R}^3$.

The theoretical role for $\mathcal{R}(\mathbb{R}^3)$ was introduced into geometric computing to correct the unexpected output seen from combinatorial operations on geometric sets [**148**]. For instance, consider the two dimensional illustration shown in Figure 4. The original operands of A and B are indicated in Figure 4(a). The unexpected output is shown in Figure 4(b), where the expected result would have been what is shown in Figure 4(c).

The phenomenon shown in Figure 4(b) was informally described as "dangling edges" [**173**]. The formalism that was proposed to eliminate this behavior was that geometric combinatorial algorithms should accept only regular closed sets as input and then execute the Boolean operations of meet, join and complementation on these operands, thereby creating only regular closed sets as output [**170**].

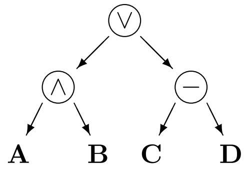


FIGURE 5. Tree for $(\mathbf{A} \wedge \mathbf{B}) \vee (\mathbf{C} - \mathbf{D})$.

The intent was to eliminate "dangling edges" and, in principle, this should have been sufficient.[1] However, each operand also has a geometric representation that depends upon the approximation methods used to compute the results. This additional subtlety raises issues in both theory and computation.

An earlier survey on topology in computer-aided geometric design [**139**] is recommended as introductory material for topologists. The texts [**96**, **136**] discuss the integration of computational geometry, shape modeling and topology.

3.2.1. *Theory versus computation.* One elegant computational representation for the combinatorial operators is to assign each object a symbol and then to indicate operations in a tree referencing those symbols. For instance, such a tree structure could be as depicted in Figure 5.

At this level of abstraction, the mathematical theory and the computational representation are completely consistent, and this representation became known as Constructive Solid Geometry (CSG). Difficulties arose in instantiating the basic geometric information that is represented by the operands at the leaf nodes and, sometimes, in computing geometric representations at the internal nodes of the tree. In CSG, the leaf nodes are restricted to a small set of specific geometric objects, known as primitives. A typical collection of primitives might consist of spheres, parallelepipeds, tori and right circular cylinders. The critical geometric algorithm underlying each Boolean operation is the pairwise intersection between the operands.

As the boundary of each of these primitives can be represented by linear or quadratic polynomials, the needed intersection between each pair of primitives was relatively simple and numerically stable, for most cases considered, although specific intersections could be problematic. For instance, suppose two cylinders of identical radius and height were created and then positioned so that the bottom of one cylinder was coincident with the top of the other cylinder. This special case was specifically considered in most intersection algorithms and could usually be processed without problem. However, if one then rotated the top cylinder a fraction of a degree about its axis (so that the planar coincidence remained intact) many software systems would fail to produce any output for this problem,

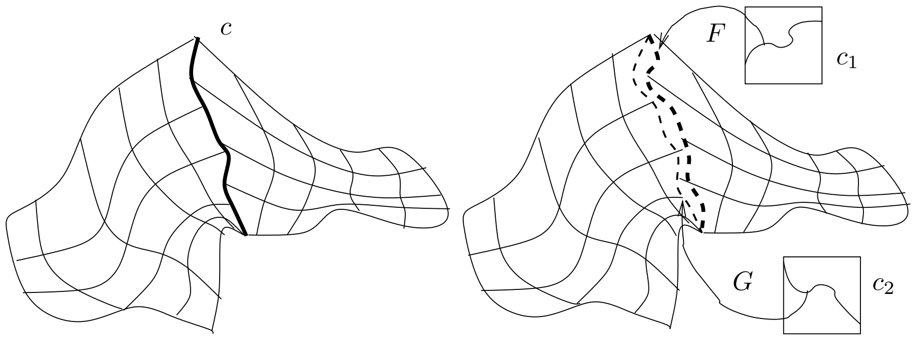
[1] The subtraction operation between two sets, shown as $A - B$ in Figure 4, is not *specifically* a Boolean operation. However, the use of $A - B$ should be understood to be conveniently shortened notation equivalent to the operations $A \wedge B'$, where B' represents the standard Boolean operation of complementation on the operand B.

sometimes even causing a catastrophic program failure. This particular problem became a celebrated test case and most systems developed *ad hoc* methods to solve this cylindrical intersection problem. Yet, this was just avoiding the more serious issue of the fragile theoretical foundations for many intersection algorithms. People using CSG systems became sensitive to their limitations and continued to use them effectively by avoiding these challenging circumstances, although the work-arounds were often tedious to execute.

The imperative, largely initiated by the aerospace and automotive industries, to model objects using polynomials of much higher degree than quadratic created a movement away from CSG systems. The alternative format was to represent compact elements of $\mathcal{R}(\mathbb{R}^3)$ by their boundaries, and this became known as the "boundary representation" approach, or "B-rep" for short. This has become the dominant mode today. Again, within this clean conceptual overview, the realities of computation pose some subtle problems. In most industrial practice, the modeling paradigm was further restricted so that the boundary of an object was a 2-manifold without boundary. However, it was difficult to create computer modeling tools that could globally define 2-manifolds without boundary, though there existed excellent tools for creating subsets of these 2-manifolds. For example, computational tools for creating splines were becoming prevalent [**141**]. Again, in principle, if each such spline subset was created with its boundaries, then the subsets could be joined along shared boundary elements to form a topological complex [**95**] for the bounding 2-manifold without boundary.

The inherent computational difficulty was to create separately two spline patches, each being a manifold with boundary, so that the corresponding boundary curves were identical and could be shared exactly between the patches. In some situations, algorithms for fitting spline patches were used successfully. In other cases, patches have been enlarged slightly and intersected so as to obtain improved fits. Indeed, such intersections are well-defined in pure mathematics, but, again, approximation in computation poses subtle variations from that theory, as described in the next section on pairwise surface intersection.

3.2.2. *Subtleties of pairwise spline surface intersection.* It is well known that unwanted gaps between spline surfaces or self-intersections within intended manifolds often appear as artifacts of various implemented intersection algorithms [**74**]. The mismatch between approximate geometry and exact topology has historically caused reliability problems in graphics, CAD, and engineering analyses, drawing the attention of both academia and industry. The severity of the problem increases with the complexity of the geometric data represented, both from high-degree nonlinearity and from the intricate interdependence of shape elements that should, but do not, fit together according to the specified topological adjacency information.



(a) Conceptual view (b) Computational reality

FIGURE 6. Spline surface interection

The conceptual view of these joining operations is illustrated in Figure 6(a), with an intersection curve[2] illustrated as a single curve. But this image only exemplifies the idealized, exact intersection curve. For practical computations, an approximation of the intersection set is often created [**85**, **86**] and, in many systems, an intersection curve will be approximated twice. These two approximations are created corresponding to each of the spline functions, denoted as $F\colon [0,1]^2 \to \mathbb{R}^3$, and $G\colon [0,1]^2 \to \mathbb{R}^3$, whose images are the surfaces being intersected. Specifically, a spline curve, denoted as c_1, is created so that $c_1 \subset [0,1]^2$ and the image of c_1 by F, denoted as $F(c_1)$ approximates the intersection curve, (with similar meaning given to $c_2 \subset [0,1]^2$ and $G(c_2)$). It is virtually certain that those approximations, $F(c_1)$ and $G(c_2)$, will not be exactly equal in $\mathbb{R}^3$, as shown in Figure 6(b).

The mismatch between concept and reality discussed above, can cause ambiguity, as the intersection representation is sometimes considered as a unique set, from the conceptual view, and at other times as two approximating sets, from an algorithmic view.

3.2.3. *Error bounds for topology from Taylor's theorem.* First, we present the Grandine–Klein (GK) intersection algorithm [**86**]. Referring to Figure 6, we note that the GK algorithm bases its error bounds on well-established numerical techniques in differential algebraic equations (DAE). While these DAE techniques provide rigorous error bounds, these bounds are expressed within the parameter space $[0,1]^2$, which serves as the domain of the spline functions (indicated as F and G, above). The code implementing the GK algorithm then has an interface that allows the user to specify an upper bound ϵ for the error within parameter space and the algorithm provides guarantees for meeting this error bound. However, the typical end user is often not fully aware of the details of the parametric definitions

[2] We focus on the generic case of an intersection curve, although isolated points and coincident areas can also arise, with similar complications.

of F and G, so selection of this parametric space error bound has often relied upon heuristics. It would be more convenient for the user to be able to specify an error bound within $\mathbb{R}^3$. One accomplishment within the I-TANGO [**137**] project has been to demonstrate a mathematical relation [**133**] between the error bounds in $\mathbb{R}^3$ and $[0,1]^2$, following from a straightforward application of Taylor's Theorem in two dimensions. The conversion between these error bounds has been implemented in a preprocessing interface to the GK algorithm and this new interface has been tested to be reliable, efficient and user-friendly.

Using the notation from Figure 12 for the spline function F, Taylor's Theorem provides a bound on the error of F evaluated at a particular point (u,v) versus F evaluated at a point (u_0,v_0), where (u,v) and (u_0,v_0) are within a sufficiently small neighborhood. This sufficiently small neighborhood will have radius given by the value in the parametric domain $[0,1]^2$ which was denoted as ϵ in the previous paragraph. Then it follows [**133**], with $\|\cdot\|$ being any convenient vector norm, that

$$\|F(u,v) - F(u_0,v_0)\| \leq \epsilon M$$

for any M satisfying

$$\left\|\frac{\partial F}{\partial u}(u^*,v^*)\right\| + \left\|\frac{\partial F}{\partial v}(u^*,v^*)\right\| \leq M,$$

for some point $[u^*,v^*]$ on the line segment joining $[u,v]$ and $[u_1,v_1]$.

For the single spline F, let $\gamma(F)$ be an upper bound for the acceptable error in $\mathbb{R}^3$ between the true intersection curve c and one of its approximants $F(c_1)$. In order to guarantee that this error is sufficiently small, it is sufficient that $\epsilon M \leq \gamma(F)$, where an upper bound for M can be computed by using any standard technique for obtaining the maximums of the partials $\frac{\partial F}{\partial u}$ and $\frac{\partial F}{\partial v}$. For G, a similar relation between $\gamma(G)$ and ϵ exists.[3]

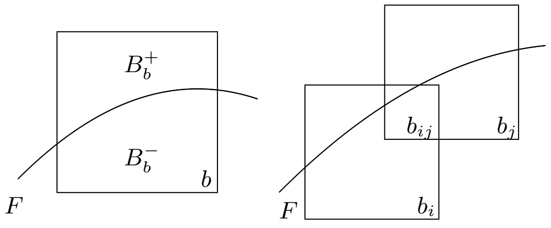
Then it is clear that a neighborhood can be defined that contains the true intersection curve c and both of its approximants. Let $N_{\gamma(F)}(F(c_1))$ be a tubular neighborhood of radius $\gamma(F)$ about $F(c_1)$, where c_1 has been generated from the GK intersector to satisfy the inequality presented in the previous paragraph. Similarly, define $N_{\gamma(G)}(G(c_2))$. Then, let $N(c) = N_{\gamma(F)}(F(c_1)) \cup N_{\gamma(G)}(G(c_2))$.

It is clear that $N(c)$ is a neighborhood of c, which contains both of its approximants, $F(c_1)$ and $G(c_2)$. However, there is both a theoretical and computational limitation to this approach.

- There is no theoretical guarantee that either approximant is topologically equivalent to c, and
- Any practical computation of $N(c)$ would depend upon an accurate computation of the set $N_{\gamma(F)}(F(c_1)) \cap N_{\gamma(G)}(G(c_2))$, which is likely to be as difficult as the original computation of $F \cap G$.

While the above bounds are often quite acceptable in practice to compute a reasonable approximant, further research has been completed into alternate methods

[3] This error bound assumed that the error due to algorithmic truncation within the numerical DAE methods dominated any other computational errors.



(a) Condition **C2** (b) Condition **C3**

FIGURE 7. 2D versions of properties of interval solids

to give guarantees of topological equivalence within a computationally acceptable neighborhood of the intersection set, as reported in the next subsection.

3.2.4. *Integrating error bounds and topology via interval solids.* Recent work by Sakkalis, Shen and Patrikalakis [**153**] emphasized that the numeric input to any intersection algorithm has an initial approximation in the coordinates used to represent points in $\mathbb{R}^3$, leading to their use of interval arithmetic [**136**]. The basic idea behind interval arithmetic is that any operation on a real value v is replaced by an operation of an interval of the form $[a, b]$, where $a, b \in \mathbb{R}$ and $a < v < b$. The result of any such interval operation is an interval, which is guaranteed to contain the true result of the operation on v. This led naturally to the concept of an *interval solid* and some of its fundamental topological and geometric properties were then proven, as summarized below.

Throughout this section, a *box* is a rectangular, closed parallelepiped in $\mathbb{R}^3$ with positive volume, whose edges are parallel to the coordinate axes.[4] Let F be a nonempty, compact, connected 2-manifold without boundary. Then the Jordan Surface Separation Theorem asserts that the complement of F in $\mathbb{R}^3$ has precisely two connected components, F_I, F_O; we may assume that F_I is bounded and F_O is unbounded. Let also $\mathcal{B} = \{b_j : j \in J\}$ be a finite collection of boxes that satisfies the following conditions:

C1: $\{\text{int}(b_j) : j \in J\}$ is a cover of F.

C2: Each member b of $\mathcal{B}$ intersects F generically; that is, $b \cap F$ is a nonempty closed disk that separates b into two (closed) balls, B_b^+ and B_b^-, with B_b^+, (B_b^-) lying in $F_I \cup F$ $(F_O \cup F)$, respectively.

C3: For any $b_i, b_j \in \mathcal{B}$, let $b_{ij} = b_i \cap b_j$. If $\text{int}(b_i) \cap \text{int}(b_j) \neq \emptyset$, then b_{ij} is also a box which satisfies **C2**.

Notice that condition **C2** indicates that every $b \in \mathcal{B}$ intersects F in a natural way (see Figure 7).

The following result summarizes several previously appearing results, where a solid is defined to be a nonempty compact, regular closed subset of $\mathbb{R}^3$.

[4] Enclosures other than boxes are quite possible and this is a subject of active research.

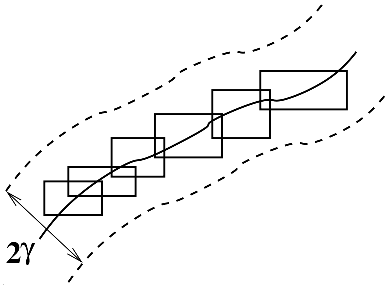


FIGURE 8. 2D version of proper subset condition

Theorem 3.1 ([**153**, Corollary 2.1, p. 165]). *If F is connected and $\mathcal{B}$ satisfies* **C1**–**C3**, *then $F \cup \bigcup_{j \in J} b_j$ is a solid.*

Bisceglio, Peters and Sakkalis [**151**, **152**] have recently given sufficient conditions to show when the boundary of an interval solid is ambient isotopic to the well-formed solid that it is approximating, as described in the following theorem. To do so, they define a parameter, denoted here as, γ, which is based upon curvature and critical values of an energy function. This value of γ then permits the definition of non-self-intersecting tubular neighborhoods about the original object for all values of $r < \gamma$, when r is a positive number for a constant radius tubular neighbhorhood. For a positive number δ, define the open set $F(\delta) = \{x \in \mathbb{R}^3 : D(x, F) < \delta\}$, where $D(x, F) = \inf\{d(x, y) : y \in F\}$, with d being the Euclidean metric in $\mathbb{R}^3$.

Theorem 3.2. *Let F be a connected 2-manifold without boundary. For each $\epsilon > 0$, there exists δ, with $0 < \delta < \gamma$ so that whenever a family of boxes $\mathcal{B}$ satisfies conditions* **C1**–**C3**, *and for each b of $\mathcal{B}$, b is a subset of $F(\delta)$* (see Figure 8) *then, for $S = F \cup F_I$ and $S^{\mathcal{B}} = S \cup \bigcup_{j \in J} b_j$, the sets F and $\partial S^{\mathcal{B}}$ are ϵ-isotopic with compact support. Hence, they are also ambient isotopic.*

The quoted theorem depends upon results from Bing's book on PL topology [**22**, p. 214], and related literature [**103**], as is explained in full [**151**, **152**]. The proof shows that normals to F do not intersect within the constructed tubular neighborhood, as is illustrated by the depiction of its planar cross-section in Figure 8.

If the boxes containing the true intersection curve can be made sufficiently small so that each such box fits inside $F(\rho)$, then the resulting intersection neighborhood will contain an object that is both close to the true solid and is ambient isotopic to it. Considerable success in meeting these constraints has already been achieved [**136**, **152**, **153**].

4. Correctly embedded approximations for graphics & applications

Papers on tolerances in engineering design [**31**, **32**, **166**] raised the issue of rigorous proofs for the preservation of topological form in geometric modeling,

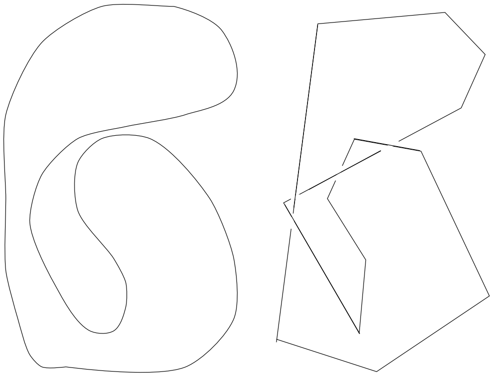


Figure 9. Nonequivalent knots

but these papers did not specifically propose ambient isotopy as a criterion. The class of geometric objects considered was expanded appreciably by theorems for ambient isotopic perturbations of PL simplexes and splines [**14**–**16**].

As an elementary example, there is an exact computational representation of a unit circle centered at the origin, as $x^2 + y^2 = 1$. However, as soon as one goes to create a computer graphics image of this circle, some approximation is needed. The ultimate display on the screen is to 'turn on' a collection of pixels, each being some very small rectangle. If these pixels are sufficiently small and the approximation is sufficiently fine, then the user perceives a reasonable image of a circle. This has many parallels to a human rendition of a circle, such as a pen and ink image that approximates a circle. The criterion for success is largely subjective, though it has been successfully codified in standard algorithms for this simple case of the circle [**77**]. This technique does generalize to more difficult geometric shapes which also have nice differentiable properties [**37**], but there remain difficulties in the prevalent approximation paradigms, as will be discussed further, here.

However, there is a crucial distinction between the use of such images in classical mathematics and in computer science. The adage in pure mathematics is that 'A picture is not a proof.' Rather, the use of illustrations is meant to guide discovery and intuition in order to lead to formal proofs. The situation in computer science is quite different. Namely, the focus is upon the definition and properties in creating specific algorithms to work on particular abstract data types. Here, the data type is the equation of a circle, but this representation is then approximated for graphics rendering. So far, this offers little distinction to the classical case. However, the output of this approximation algorithm may often be used by another algorithm. The approximation becomes the object of interest. This could be translating the circle to another position or determining its circumference. Both these operations are quite successful for the circle.

Indeed, even at the graphics display level, the concern for a 'properly representative approximation' should not be dictated solely by subjective criterion, as

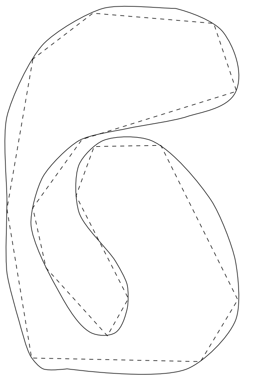


FIGURE 10. Ambient isotopic approximation

can be shown in the following knot approximation example, which summarizes a published example [**13**].

Many geometric approximation algorithms offer no guarantees about the topology of the output. Sometimes it is guaranteed that the output is homeomorphic to a desired manifold [**10**]. Indeed, in the simple circle example, essentially any reasonable PL approximation of the cirle will be homeomorphic to it. However, in graphics, any 3D image is projected onto a 2D display. One asks if this composition of functions will necessarily lead to a homeomorphic image. The answer can be easily shown to be 'no' and supports the argument for a stronger form of topological equivalence.

We argue here that a guarantee of homeomorphism is insufficient for many of the applications for which the algorithms are designed. Rather, examples are given for preferring a stronger equivalence relation based upon ambient isotopy.

Definition 4.1. If X and Y are subspaces of $\mathbb{R}^3$, then X and Y are *ambient isotopic* if there is a continuous mapping $F\colon \mathbb{R}^3 \times [0,1] \to \mathbb{R}^3$ such that for each $t \in [0,1]$, $F(\cdot,t)$ is a homeomorphism from $\mathbb{R}^3$ onto $\mathbb{R}^3$ such that $F(\cdot,0)$ is the identity and $F(X,1) = Y$.

For other fundamental terms, the reader is referred to the text [**94**].

Figure 9 shows an unknot, and its homeomorphic, *but non-isotopic* PL approximant, which is the knot with 4 crossings, known as 4_1^m. An improved approximation is shown in Figure 10.

All end points of the line segments in the approximation are also points on the original curve. In response to the example of Figure 9, a theorem was published for ambient isotopic PL approximations of 1-manifolds [**119**], with an illustrative outcome shown in Figure 10. The proof utilizes 'pipe surfaces' from classical differential geometry [**128**].

Although any two simple closed planar curves are ambient isotopic, this knotted curve as an approximant to the original unknot would be undesirable in many circumstances, such as graphics and engineering simulations [**15**]. For instance, projected images of this approximation could have self-intersections, whereas the original curve had none.

There is a related study of curves, comparing them to α-shapes [**69**] via ambient isotopies [**150**]. Recent work in support of molecular modeling appears in the doctoral thesis of E.L.F. Moore and related publications [**129–131**].

Other problems arise for surfaces (2-manifolds) in three dimensions. Some algorithms compute a triangulated surface C to approximate the boundary F of a closed, finite volume, with a guarantee that C is homeomorphic to F [**11**, **12**]. It is well known that this does *not* guarantee that the complement of C, denoted as $\mathbb{R}^3 \setminus C$, is homeomorphic to the complement of F, $\mathbb{R}^3 \setminus F$, meaning that there is no guarantee that F and C are equivalently embedded in $\mathbb{R}^3$. An ambient isotopy between C and F, on the other hand, provides such a guarantee.

The class of PL surfaces presents another domain in which topological guarantees are desirable. Even guaranteeing that the common *edge contraction* operator produces an object homeomorphic to its input requires some care for simplicial complexes [**52**]. Preservation of genus during approximation by a polygonal mesh [**171**] also requires considerable care.

Recent theorems [**4**, **5**, **13**] prove approximation techniques that preserve ambient isotopy over an important subclasses of 2-manifolds, covering cases both with and without boundary. The role for ambient isotopy has been recognized by the computer animation research community [**80**].

1114? **Question 4.1.** *Is ambient isotopy the appropriate topological equivalence relation for computational topology in computer graphics and animation?*

Question 4.2. *What geometric approximation algorithms can capture the topological equivalence needed in computer graphics and animation?*

1115? **Question 4.3.** *Are the known algorithms for ambient isotopic of parametric curves optimal with respect to performance and space requirements?*

1116? **Question 4.4.** *Are the known algorithms for ambient isotopic of parametric surfaces optimal with respect to performance and space requirements?*

Considerable work on isotopies in approximation has appeared, ranging over applications from computer graphics, geometric modeling and surface reconstruction [**38–40**].

5. The role for differentiability

Although computational topology is a relatively new discipline [**21**, **172**], it has grown and matured rapidly partially because of its increasing importance to many vital contemporary applications areas such as computer aided design and manufacturing, (CAD/CAM), the life sciences, image processing and virtual reality. It is leading to new techniques in algorithm and representation theory. These applications are evoking new connections between mathematical subdisciplines such as algebraic geometry, algebraic topology, differential geometry, differential topology, dynamical systems theory, general topology, and singularity and stratification theory [**1**–**3**]. The tender age of computational topology renders it fertile ground for a wide variety of challenging open problems—many of fundamental importance. While the primary focus of this book and, of course, this chapter is upon problems in general topology, the integrative nature of computational topology is expressed here with some attention to the role of differentiability.

5.1. Introduction. Computational geometry preceded computational topology as an indispensable theory and practice for solving difficult problems that have arisen in CAD/CAM and other contexts that rely on computationally powerful methods for analysis and accurate representation of various objects and configurations. On the other hand, computational topology has only considerably more recently risen to prominence in such applications [**74**]. The difference between these two disciplines is roughly analogous to the difference between geometry and topology, and can be rather effectively illustrated in the following terms: Whereas computational geometry is concerned essentially with algorithmic (and *a fortiori* computer implementable) methods for analyzing and producing representations of geometric objects that are close—usually in some Whitney-like (piecewise) C^2 sense—a primary focus of computational topology is to guarantee algorithmically that a computer generated representation of an object is equivalent to the actual object in an appropriate topological sense. In essence, computational geometry is concerned with insuring the (differential geometric) closeness of the representation of an object to the original, while computational topology takes care of the topological consistency of the rendering.

The importance of computational topology cannot be overestimated in certain contexts and applications—many of which have achieved significant prominence in the last few years. For example, suppose one wants to produce a computer generated representation of a water glass to be used in an automated manufacturing process. The glass can be viewed in ideal form as a smooth surface in space with a circular boundary, thus rendering it an object in a standard differential geometry or topology category. An algorithm can readily be found that produces a representation that is as close as desired (in some suitable Whitney-type topology) to the designed glass, but still have in it a very small hole. This may be considered satisfactory from the perspective of computational geometry, but certainly not from the computational topology viewpoint, and the result obviously would lead to shortcomings in the manufactured article.

In this section we shall identify several outstanding problems in computational (differential) topology—all of which are of a rather fundamental nature—and we also shall provide the necessary context and background for an appreciation of these problems, along with some insights that should prove helpful in their resolution. As computational topology is still an emerging discipline and is largely unknown to many in the computer aided geometric design, computer science, and mathematics communities, we shall present a brief outline of the elements of computational differential topology in Subsection 5.2, a description of the problem of identifying and classifying those objects in a category associated with computational differential topology in Subsection 5.3 and algebraic duals of previous problems now expressed as issues in isomorphism type in Subsection 5.4. In particular, we treat in Subsection 5.4 those that possess a complete set of effectively (algorithmically) computable topological invariants, i.e., those geometric objects that have sufficiently many algorithmically computable invariants to completely determine their isomorphism classes in an appropriate topological category.

5.2. Elements of computational differential topology. One unmistakable sign of a mature mathematical or scientific subdiscipline is the establishment and general acceptance of well defined mathematical categories that characterize and circumscribe the field. Such categories have yet to be universally embraced in the computational topology community, so we shall first describe the categories in which we shall work in order to frame the problems to be posed in this paper. This Subsection has its own Subsections 5.2.1 discussing the categorical structures needed; 5.2.2 raising the issue of shape equivalence within these categories and 5.2.3 emphasizing the interplay between topology and algorithms.

5.2.1. *Categories.* The sets of interest in computational topology are geometric objects in an Euclidean space, usually having certain differentiability properties, but they need not and should not be restricted to manifolds. Examples such as the locus of $x^2 + y^2 - z^2 = 0$ in $\mathbb{R}^3$ and geometric objects with self-intersections show that we need to include varieties. One approach to describing the objects in an appropriate category is to introduce special varieties (s-varieties) having the property that there are at most finitely many local regular (topological manifold) branches at each of the singular points [**26**, **27**]. However, a more efficient way to describe the objects in the computational topology categories is to employ *Whitney regular stratifications* [**19**, **27**, **28**, **41**, **84**, **123**, **168**, **178**]. First we fix an Euclidean space $\mathbb{R}^N$ to serve as the ambient space for the geometric objects and an order of differentiability k $(0 \leq k \leq \infty)$.

Definition 5.1. For a given Euclidean space $\mathbb{R}^N$ and order of differentiability $0 \leq k \leq \infty$, a computational differential topology object, denoted as cdt_N^k, is a subset V of $\mathbb{R}^N$ that can be represented in the form

$$V = M_1 \cup M_2 \cup \cdots \cup M_s, \tag{5.1}$$

where the collection $\mathfrak{S} := \{M_i : 1 \leq i \leq s\}$ is a Whitney regular stratification of V. This stratification is comprised of a finite disjoint set of strata M_i, which are open or closed C^k submanifolds of $\mathbb{R}^N$, called the strata of the stratification, and the strata have dimensions that can range from 0 (points) to N (open solid regions). The dimension of V in cdt_N^k is defined as $\dim V := \max\{\dim M_i : M_i \in \mathfrak{S}\}$.

Note that a cone is in cdt_3^∞, as is a closed cube. Since we shall be concentrating in this paper mainly on geometric objects that have some differential structure, most of our attention shall be directed to cases where $k \geq 1$.

We now have suitable objects for our categories, so it naturally remains to define the appropriate morphisms. It is clear that the more usual choice leading to homeomorphic or diffeomorphic equivalence will simply not do. For example, a circle S^1 and a smooth trefoil T knot embedded in $\mathbb{R}^3$ are obviously C^∞-diffeomorphic, 1-dimensional submanifolds, but can certainly not be viewed as equivalent in any reasonable computational topology sense since they are not equivalent as embeddings in the ambient space $\mathbb{R}^3$. In particular, the knot group for the circle is $\pi\left(\mathbb{R}^3 \setminus S^1\right) = \mathbb{Z}$, while the knot group for the trefoil knot $\pi\left(\mathbb{R}^3 \setminus T\right)$ is the group with two generators α and β and one relation, $\alpha\beta\alpha = \beta\alpha\beta$, where $\pi(X)$ denotes the fundamental group of a topological space X. Therefore, morphisms must be equivalent in some sense as embeddings in the ambient space, as well as having certain differentiability properties. The next definition attends to these requirements.

Definition 5.2. A morphism between two objects V and W in cdt_N^k is an embedding (in the topological sense) $\Phi\colon \mathbb{R}^N \to \mathbb{R}^N$ satisfying the following properties:

(i) $\Phi(V) \subseteq W$.

(ii) The restriction $\Phi_{|V}$ of Φ to V is of class C^k.

With this we have the last piece necessary for the definition of our computational topology categories for objects embedded in an Euclidean space $\mathbb{R}^N$:

Definition 5.3. For a given Euclidean space $\mathbb{R}^N$ and order of differentiability $0 \leq k \leq \infty$, the computational differential topology category, denoted as CDT_N^k, is comprised of all the objects in cdt_N^k as in Definition 5.1, and the morphisms as in Definition 5.2, with the usual composition of morphisms.

Observe that according to this definition, two objects V and W in CDT_N^k are isomorphic, denoted as $V \approx_N^k W$, iff there is a homeomorphism $\Phi\colon \mathbb{R}^N \to \mathbb{R}^N$ such that $\Phi(V) = W$, and the restrictions of Φ to V and its inverse Φ^{-1} to W are both of class C^k. We remark that in most cases when the ambient space and differentiability class are fixed, we simplify the above notation by omitting the subscript and superscript in the isomorphism notation, so that we shall simply write $V \approx W$. In the sequel we shall, for convenience, indulge in the harmless abuse of notation of referring to both objects and morphisms as being members of the category CDT_N^k rather than distinguishing between the set of objects and set of morphisms comprising this category.

Isomorphism in the categories CDT_N^k (which is sometimes referred to as embedding equivalence [**149**]) is obviously more restrictive than homeomorphic equivalence in the standard topological category TOP. More specifically, in addition to the usual homeomorphism type invariants such as homotopy, cohomotopy, homology, and cohomology that one needs to consider for equivalence in TOP, one needs also to verify the invariance of additional quantities such as linking numbers to verify equivalence in the computational differential topology categories. For future reference, we denote isomorphism in the TOP category as

$$V \overset{\mathrm{h}}{\approx} W. \tag{5.2}$$

Topological equivalence by isotopy [**13**, **14**, **16**, **23**, **139**, **151**] is stronger than the isomorphic equivalence given in Equation 5.2, as has already been introduced in Definition 4.1. We remark here that for the case of smooth knotted and unknotted circles in $\mathbb{R}^3$, standard knot equivalence, ambient isotopy, and isomorphism in CDT_3^0 are all equivalent to one another [**91**, **92**, **122**].

One of the basic goals in computational topology is to create computer generated procedures for obtaining representations of objects having the same shape—at least in some acceptable approximate sense—as a given geometric object. This obviously begs the question of what is meant by shape, a question that we address in the next subsection.

5.2.2. *Shape of geometric objects.* What does it mean to say that two objects, V and W in CDT_N^k have the same shape? Naturally, to have the same shape, V and W ought to at least be isomorphic in the computational topology category, but intuition certainly requires more. A suitable definition is the following:

Definition 5.4. The objects V and W in CDT_N^k have the same shape if there exists an isomorphism $\varphi\colon V \to W$ that is a scaled C^k-isometry in the following sense: There exists a constant $c > 0$ such that $c^{-1}\varphi$ is an isometry. More particularly, recall that for φ to be an isomorphism in CDT_N^k it must be extendable to a homeomorphism $\Phi\colon \mathbb{R}^N \to \mathbb{R}^N$. Accordingly the definition requires that the restriction of Φ to V (which is φ) must be a C^k map such there exist a $c > 0$ and an isometric C^k-embedding $\psi\colon V \to \mathbb{R}^N$ (in the metric induced on V by the Euclidean metric on $\mathbb{R}^N$) with $\Phi(x) = c\psi(x)$ for all $x \in V$. We denote this property of having the same shape by $V \equiv_N^k W$, and omit the subscript and superscript for simplicity whenever the context is clear.

Computational representations of geometric objects—no matter what type of format is used to describe the rendered object—usually involve some approximation error, which necessitates the use of the following definition, or something of the same sort, for computational topology applications.

Definition 5.5. Given $\epsilon > 0$, we say that V and W in CDT_N^k have the same shape $(\operatorname{mod} \epsilon)$ if they are isomorphic in this category via $\varphi\colon V \to W$, and there are a positive number c and an isometric C^k-embedding $\psi\colon V \to \mathbb{R}^N$ such that φ is ϵ-close to $c\psi$ in the Whitney C^k-topology, which essentially means that derivatives of all orders less than or equal to k of φ and $c\psi$ differ by less than ϵ (in the appropriate operator norm) over all of V [**19**, **84**, **134**]. Having the same shape $(\text{mod } \epsilon)$ is denoted as $V \equiv_N^k W$ $(\text{mod } \epsilon)$, where as usual we shall suppress the subscript and superscript when the context is clear.

We now are in possession of all the notation that we need to formulate the efficient approximation problem of computational differential topology, which we attend to in the succeeding subsection.

5.2.3. *The efficient approximation problem.* With the notation, it is simple to explain—at least in general terms—the nature of the essential problem confronting computational topologists. It begins with a given prototype object V_0 in CDT_N^k, which must be represented by computer generated means that are based upon an algorithm $\mathcal{A}$. The word 'given' here is somewhat of a misnomer that requires very broad interpretation: The prototype object may be defined exactly in terms of equations, or it may be a completely developed model of a geometric object, or represented by data sampled from an existing physical object such as a statue or building, or—in the worst case scenario—may be only partially and imprecisely known simply in terms of representative data, such as point-clouds, sampled according to some scheme [**161**].

An algorithm for representing and analyzing a geometric object with computational topology constraints should include an algorithmic subroutine for verifying that the computed object has the same isomorphism type as the given object—assuming that this much is known about the object to be represented. If we have only incomplete topological knowledge of the prototype object, an algorithm designed to produce computer generated representations, say at various levels of accuracy, should at least be capable of verifying that the isomorphism type remains constant as the accuracy is refined. When such an algorithm is available, such a constant 'limit' may serve as a good educated guess of the actual isomorphism type of the partially known prototype object. The following notion is useful in the investigation of such questions.

Definition 5.6. Let V_0 be a given object in CDT_N^k and let V be another such object. Then the isomorphism type of V is said to be V_0-*decidable* if there exist an algorithm $\mathcal{A}$ to determine if $V \approx V_0$. Such an algorithm is called a (V_0, V)-decider.

This brings us to the overarching focal point of any complete investigation of a problem in computational differential topology, which addresses both the mathematical and computer science aspects involved.

Efficient Approximation for Computational Differential Topology. *Given a prototype object V_0 in the category CDT_N^k, construct an algorithm $\mathcal{A}$ to be used for obtaining a computer generated representation V (in CDT_N^k) of V_0, which has the following properties:* (a) *For each sufficiently small positive ϵ, the algorithm generates a representation $V(\epsilon)$ of V_0 and includes a subalgorithm that is a $(V_0, V(\epsilon))$-decider;* (b) $V(\epsilon) \equiv V_0 \pmod{\epsilon}$ *for all such ϵ; and* (c) *the algorithm is optimally efficient to the degree that the computational complexity of $\mathcal{A}$, denoted as $CC(\mathcal{A})$, is minimal in some reasonable sense.*

It should be noted that, although not specifically included in the above definition of the efficient approximation problem, ease of implementation with regard to producing user-friendly software based on the algorithm is also an important consideration, especially when it comes to applications.

In general, a complete solution of the efficient approximation problem as stated may be extremely difficult—or even impossible—to achieve, so simplified versions of this problem, such as those we describe in the sequel, are highly desirable and often vigorously pursued. We note that if this efficient approximation problem is viewed from a computational geometry rather than a computational topology viewpoint, one should choose the differentiability class k to be greater or equal to two, so that the representations produced are acceptable in terms of differential geometry where second derivatives manifested in curvature tensors (or differential forms) are essential elements in determining good approximations.

5.3. The identification and classification problem. The reader is sure to have observed that the efficient approximation problem as presented in the preceding sections is somewhat lacking in rigor. Moreover, as Edelsbrunner pointed out when the version above was unveiled recently, it also is deficient in scope—especially as regards the wide range of possibilities in knowledge of the prototype object, means of obtaining data from the object for the algorithm, and methods available for rendering the computational representations. These observations constitute the core of the first few open problems that we pose here.

5.3.1. *Formulation of the identification and classification problem.* In order to pose this identification and classification problem with more precision, and to introduce sufficient rigor into supporting definitions and concepts so as to articulate which problems remain open, we shall first present a more detailed version than outlined in the preceding section. To begin, we develop more precise notation concerning the computational procedures embodied in the algorithm $\mathcal{A}$ devised to produce an approximate representation $V(\epsilon)$ of the prototype geometric object V_0 in CDT_N^k for a given error bound ϵ. We emphasize here that the error bound is on the geometry, not the topology, as invariance of the isomorphism type is an essential requirement for the algorithm. The input data from V_0, which we denote as $D\,(V_0)$, may assume any one of several possible forms such as the vertex points and connection relations for the elements of a triangulation of the prototype object, a global functional representation or a set of local functional expressions arising from exact mathematical models, or an approximate nonuniform rational B-spline (NURBS) decomposition of V_0, or points forming a point-cloud sampled in a manner designed to provide a good approximation of the given object, which is often the case when V_0 is not completely known or specified.

One can already see here that there is a problem in formulating an adequate characterization of the space $\mathfrak{D}$ in which the data obtained from the prototype object resides. A good definition of this data space is required so that we can consider D as a function from (the object set of) CDT_N^k to $\mathfrak{D}$, which can be expressed as $D\colon \mathsf{CDT}_N^k \to \mathfrak{D}$. Of course, the tolerance (geometric accuracy) ϵ must also be counted as an argument of the algorithm. With the notation developed, we may now view the algorithm as a recursive map of the form

$$\mathcal{A}\colon D(\mathsf{CDT}_N^k) \times \mathbb{R}_+ \to \mathsf{CDT}_N^k \qquad (D(V_0), \epsilon) \longmapsto V(\epsilon)$$

where $\mathbb{R}_+$ is the set of positive real numbers, and $V(\epsilon)$ is a graphical rendering of a (geometric) approximation of V_0, or more precisely, an algorithm for producing a computer generated approximate representation of the prototype object. We now have a more rigorous foundation for describing the identification and classification problem.

The Identification and Classification Problem in CDT_N^k. *Devise an algorithm $\mathcal{A} = \mathcal{A}\,(D(V_0), \epsilon)$ that*

(i) *is defined for all sufficiently small positive ϵ,*
(ii) *is defined for a suitably ample domain of prototype objects V_0 in CDT_N^k,*
(iii) *produces an output $V(\epsilon) \equiv_N^k V_0 \pmod{\epsilon}$ for all ϵ for which it is defined,*
(iv) *has minimal computational complexity $CC(\mathcal{A})$ in some sense.*

The above description of the identification and classification problem, although more precise than that which was presented in preceding section, is clearly still beset with deficiencies in several respects, two of which are embodied in the following posed problems.

Question 5.1. *Modify the description of the Identification and Classification Problem in CDT_N^k so that it more rigorously and completely encompasses the wide range of methods that can be used, and is better able to express the degree to which the prototype object is known.*

Question 5.2. *Find a way of better expressing the type of representation approach that is used to produce the output object $V(\epsilon)$ in the statement of the identification and classification problem in CDT_N^k.*

Note that in a case where the isomorphism class of the prototype object V_0 in CDT_N^k is mostly or partially unknown, it will be necessary to revise the requirement (iii) to something like

(iii)′ The outputs $V(\epsilon_1)$ and $V(\epsilon_2)$ with $0 < \epsilon_1, \epsilon_2 < \epsilon$ satisfy $V(\epsilon_1) \equiv_N^k V(\epsilon_2) \pmod{\epsilon}$ for all sufficiently small ϵ.

This suggests a possible notion of persistence of isomorphism type analogous to the basic ideas used to formulate persistent homology [**68**, **72**, **184**, **185**].

Question 5.3. *Reformulate and expand (iii) in the Identification and Classification Problem in CDT_N^k to include those cases where one only has incomplete knowledge of the isomorphism type of the prototype object—perhaps along the lines of persistence of isomorphism type for sufficiently small tolerances.*

Another inadequacy of our exposition of the identification and classification problem is manifested in the imprecision of the minimality statement for computational efficiency, which naturally leads to the following question.

Question 5.4. *Revise the definition of the Identification and Classification Problem in CDT_N^k so that it includes a more precise description of the computational cost that is consistent with the most important computational concerns arising in a broad spectrum of applications of computational topology.*

Resolving this minimality definition problem is bound to be quite challenging, partly owing to the extensive array of minimality criteria available for applications, but more likely to stem from the difficulty of actually proving minimality for an algorithm in most reasonable, nontrivial senses. As algorithms developed to render approximations of geometric objects possessing only a fair degree of complexity tend to be decidedly nontrivial, verifying minimality of computational complexity tends to be rather daunting.

In addition to the properties of the algorithm $\mathcal{A}$ delineated in the identification and classification problem, it is desirable for it to continue to generate representations satisfying property (iii) or (iii)$'$ when the data $D(V_0)$ and tolerance ϵ vary slightly in an appropriate sense. When the algorithm has this additional feature, it is natural to say that it is *stable*, and this leads to another problem.

Question 5.5. *Devise a rigorous definition of stability of computational topology algorithms, and develop methods for determining whether or not such an algorithm is stable.*

It should be clear to anyone with experience in solving problems in computational topology that it might help to ameliorate the inherent ambiguity of the identification and classification problem if some of the techniques for determining isomorphism type (at least approximately) were included in the above description. Most of the methods currently employed to analyze isomorphism type involve the algorithmic computation, where feasible, of isomorphism invariants such as characteristic classes, homology groups, and cohomology rings, along with approaches based upon tubular type neighborhoods, Morse theory, Morse–Floer theory, singularity/stratification theory, and obstruction theory [**1**, **3**, **100**, **120**, **126**, **133**, **137**, **140**, **151**, **158**, **184**, **185**]. However, there also is a fairly recent spate of articles employing innovative methods from general topology, such as [**73**, **81**, **83**, **93**, **105**, **107**, **112**, **121**], that appear to be applicable to the (complete or partial) computation of isomorphism type as well.

5.3.2. *Simplification of the identification and classification problem.* Owing to the impressive advances in the realm of computational geometry over the last several decades leading to the creation of several algorithms for generating very (geometrically) accurate representations of geometric objects, and the development of new tubular neighborhood based theorems, it now appears possible to recast the identification and classification problem in the following far more tractable simplified form.

Simplified Identification and Classification Problem. *Devise an algorithm* $\mathcal{A} = \mathcal{A}\left(D(V_0), \epsilon\right)$ *that*

(i) *is defined for all sufficiently small positive* ϵ*,*
(ii) *is defined for a suitably ample domain of prototype objects* V_0 *in* CDT^k_N*,*
(iii) *produces an output* $V(\epsilon)$ *that is* ϵ*-close to and has the same homeomorphism type as* V_0 *for all* ϵ *for which it is defined,*
(iv) *has minimal computational complexity* $CC\left(\mathcal{A}\right)$ *in some sense.*

The basis for the above simplification is what has been called the *self-intersection precedes knotting principle* (SIPKP), which can be explained in the following way for compact objects V in CDT^k_N. Owing to the compactness, all of the strata in the regular stratification (Equation 5.1) of V_0 have compact closure. Each closed stratum has an arbitrarily thin, relatively compact tubular neighborhood, and the open strata also can be shown to have arbitrarily thin, relatively compact tubular-like neighborhoods. A tubular-like neighborhood for an open stratum has the form of a standard tubular neighborhood joined to open neighborhoods of the ends of the manifold, very much like the construction for manifolds-with-boundary employing boundary collars in [**4**, **5**]. Taking the union of these tubular and tubular-like neighborhoods for all the strata, we have an arbitrarily thin tubular-like neighborhood U. Then we can use an existing computational geometry algorithm to generate an approximation $V(\epsilon)$ contained in U. When the distance between images of a homeomorphism differ by no more than ϵ, some sufficient conditions are known to extend these homeomorphisms to ambient isotopies [**103**]. This known proof avoids the self-intersections mentioned, leading to the following open problem as to how far this technique can be extended.

Question 5.6. *Prove the SIPKP, or provide a counterexample.* 1117?

It goes almost without saying that there are obvious versions of Problems 5.1–5.6 for the simplified identification and classification problem, and these too are open problems of fundamental importance in computational topology.

5.4. Decidability of isomorphism type. The discussion of the identification and classification problem and a simplified version of it in the previous section raises the question of just what types of objects in CDT^k_N are amenable to algorithmic determination of their isomorphism types. We shall focus on this question in this section (assuming some familiarity with the basics of differential topology and such related fields as singularity and stratification theory as can be found in [**19**, **27**, **52**, **59**, **84**, **90**, **94**, **122**, **123**, **125**, **134**, **149**, **163**, **168**, **174**, **178**]), and will find it convenient to employ the following definition.

Definition 5.7. Let C be an arbitrary category, and suppose that X is an object in this category. If there is an algorithm for determining the isomorphism class of X, we say that X is C-*decidable*.

Bearing this in mind, we shall concentrate on identifying the properties that render a geometric object (embedded in an Euclidean space) decidable in the relevant categories for computational differential topology. To establish the overall theme of this section, we shall first summarize everything in one overarching problem, and then proceed to break this up into more manageable pieces. This unifying problem may be phrased in the following manner.

Unifying Topological Decidability Problem. *Determine all compact objects in* CDT^k_N *that are*

(a) *TOP-decidable,*
(b) CDT^k_N*-decidable,*

and determine the algorithm of minimum computational complexity capable of deciding the isomorphism type in each case.

We shall begin with compact submanifolds and submanifolds-with-boundary in CDT^k_N, with $1 \leq k$, as they are typically easier to classify in terms of the categories of interest here, namely CDT^k_N and TOP.

5.4.1. *Decidability of compact submanifolds.* In our discussion, we shall proceed in the order of increasing dimension N of the ambient Euclidean space. If $N = 1$, any compact submanifold, denoted as M, is *closed* (because by definition it has an empty boundary, i.e., $\partial M = \varnothing$). Hence, M is particularly simple, a finite set of points in the zero-dimensional case. There are no closed compact submanifolds of $\mathbb{R}^1$ of dimension one (or equivalently, of codimension zero, which is the dimension of the ambient space minus the dimension of submanifold). Even if we drop the compactness assumption, decidability is a simple matter owing to the fact that every connected, open, C^1 submanifold of $\mathbb{R}^1$ of codimension zero is an open interval. The compact submanifolds-with-boundary of $\mathbb{R}^1$ are also easy to classify algorithmically in CDT^k_N, for they must be one-dimensional and comprised of finitely many disjoint closed intervals. We note from these simple examples that we may assume that the submanifolds are connected, for if not, we can simply analyze the components one-by-one.

In $\mathbb{R}^2$, the situation is also essentially trivial, with the decidability of the homeomorphism type or isomorphism type in CDT^k_N being a simple matter indeed. For example, it follows from the Jordan curve theorem and other basic principles, that every connected, closed submanifold M of codimension-1 must be equivalent to the circle S^1 in either the category TOP or CDT^k_N. Moreover this can be determined by a single effectively computable invariant, which is the condition $H_1(M, \mathbb{Z}) = \mathbb{Z}$ for the first integral homology group, or equivalently described in terms of the Euler–Poincaré characteristic as

$$\chi\,(M) = \sigma_0 - \sigma_1 = \operatorname{rank} H_0\,(M, \mathbb{Z}) - \operatorname{rank} H_1\,(M, \mathbb{Z}) = 0,$$

where σ_j stands for the number of j-dimensional simplices in a triangulation, and the rank is defined in the usual way [**122**, **135**, **163**, **184**]. Note also that if we choose an algorithm $\mathcal{A}$ based on computation of χ, we readily find that $CC\,(\mathcal{A}) = O\,(n_s)$, where n_s is the number of top (=1)-dimensional simplices in a triangulation of M, and one cannot do much better than this with respect to computational efficiency. As a matter of fact, it follows readily that both the complete and simplified identification and classification problems are completely solved for compact submanifolds of $\mathbb{R}^2$, including the establishment of computational minimality for the algorithm assuming that the prototype submanifold is completely simplicially defined in terms of triangulations.

These simple results already provide an indication of the usefulness of algebraic topology in dealing with the decidability problem. In this vein, we include the following result for future reference. It can be proved using the stratification (Equation 5.1), the C^1 triangulation theorems of Munkres [**134**], and some basic results on the effective (algorithmic) computability of homology and cohomology for finite simplicial complexes (see [**100**, **135**, **184**]).

Theorem 5.1. *Let V be a compact object in CDT^k_N $(k \geq 1)$. Then V has a finite C^1 triangulation, and the homology $H_*\,(V, F)$, cohomology $H^*\,(V, F)$, and all of the applicable characteristic classes such as the Chern, Euler, Stiefel–Whitney, and*

Pontryagin classes (possibly just for the strata) for V are effectively computable, where the coefficient ring F is the integers $\mathbb{Z}$ or the integers mod 2.

It is in $\mathbb{R}^3$ that both the isomorphism classification and the decidability problem first assume nontrivial proportions.

Compact manifolds in Euclidean 3-space: Let M be a compact, connected submanifold (possibly with boundary) in CDT^k_N with $k \geq 1$. When $\dim M = 0$, both the classification and decidability problem are trivial in both TOP and CDT^k_N. For $\dim M = 1$, things begin to get very interesting and rather difficult. If M is closed, it must be diffeomorphic to a circle, but it can be embedded in $\mathbb{R}^3$ as a very complicated knot. Decidability in TOP is virtually the same as in the 2-space case described above, so the homeomorphism type can be algorithmically determined in linear time. In CDT^k_N, the isomorphism classes correspond to knot types. It follows from [**91**, **92**] that M is CDT^k_N-decidable, but may be NP-complete. This contrast is a very effective demonstration of how much more difficult it can be to solve the complete identification and classification problem than the simplified identification and classification problem.

An embedded closed surface M, must be orientable, and an easy solution of the decidability problem follows directly from the simple and elegant classical result [**122**, **163**] that the homeomorphism and diffeomorphism types of such a submanifold are completely determined by the Euler–Poincaré characteristic

$$\chi(M) = \sigma_0 - \sigma_1 + \sigma_2 = \operatorname{rank} H_0(M, \mathbb{Z}) - \operatorname{rank} H_1(M, \mathbb{Z}) + \operatorname{rank} H_2(M, \mathbb{Z}).$$

Accordingly the problem for TOP-decidability is solvable in linear time. Again, there is a huge difference in the degree of difficulty of the TOP- and CDT^k_N-decidability problems, as one can see by considering the thin toral surface of a smoothly thickened knotted curve. Once again, M is CDT^k_N-decidable—although there seems to be no proof of this in the literature—but the computational complexity of any associated algorithm appears to be very high, and may be NP-complete.

The homeomorphism or diffeomorphism types of a compact submanifold-with-boundary M of codimension-2 in $\mathbb{R}^3$—which may be nonorientable as in the case of a Möbius strip—is completely determined by $\chi(M)$, the orientability, and the number of boundary components [**122**]. Therefore, M is TOP-decidable in linear time. On the other hand, if M is CDT^k_3-decidable, then the computational complexity of the problem is bound to be of the order of knot decidability, but otherwise appears to be unknown.

Question 5.7. *Prove*[5] *that every compact, connected, C^1-submanifold of $\mathbb{R}^3$ of* 1118?
dimension less than or equal to 2 is CDT^k_3-decidable and obtain estimates for the computational complexity of any relevant algorithms that can be used to determine isomorphism type.

A compact, connected, 3-dimensional, C^1-submanifold M of $\mathbb{R}^3$ must have a nonempty boundary ∂M. It is easy to see that if ∂M is connected, it completely

[5] All problems of providing a proof include implicitly the option of finding a counterexample.

determines M; hence, M is decidable in both TOP and CDT_3^k. An analog of this ought to be true in the case when ∂M is not connected, but this still appears to be an open problem.

1119? **Question 5.8.** *Prove that every compact, connected, C^1-submanifold of $\mathbb{R}^3$ of dimension 3 is both TOP- and CDT_3^k-decidable (or provide a counterexample), and obtain estimates for the computational complexity of any relevant algorithms that can be used to determine isomorphism type in these categories.*

Compact manifolds in Euclidean 4-space: Of course there is a far more diverse and interesting range of compact submanifolds of $\mathbb{R}^4$ than $\mathbb{R}^3$, but we shall confine our attention to just a select few of the possible types of C^1-submanifolds of dimension two or higher. Moreover, in this and the higher dimensional cases in the sequel, we shall concentrate mainly on TOP-decidability, which is associated with the simplified identification and classification problem. We observe that all closed surfaces, or compact surfaces-with-boundary, including the nonorientable ones such the Klein bottle and the projective plane, can be embedded in $\mathbb{R}^4$.

We showed above how the decidability problem for oriented compact surfaces can be easily and very efficiently solved. This is also true for the nonorientable surfaces, all of which can be realized as two-dimensional, closed submanifolds and compact submanifolds-with-boundary of $\mathbb{R}^4$. For these cases the TOP and CDT_4^k isomorphism types also are completely determined by the orientability, or lack thereof, the Euler–Poincaré characteristic, and the number of boundary components. Moreover, the isomorphism type can be computed in linear time.

To summarize compact surfaces with regard to the decidability problem: they represent the lowest dimensional nontrivial submanifolds for which the problem becomes interesting, yet is easily solvable by simple classical means expressed, modulo orientability and possible boundary components, in terms of a single invariant that is computable in linear time. As such, they are excellent illustrative examples of some of the simplest solutions that provide direction for more general cases.

The 3-sphere S^3 is the simplest closed, connected, three-dimensional, submanifold of $\mathbb{R}^4$. It has been much in the mathematical news of late owing to the excitement created by the work of Perelman [**98**] on the famous and long-standing Poincaré Conjecture, which states that a connected, simply-connected (i.e., $\pi(M) = 0$) three-dimensional manifold M having the homology of a 3-sphere must, in fact, be homeomorphic with S^3 [**132**]. Perelman's work, which relies heavily upon Hamilton's Ricci flow methods, is still being studied by the experts, and at last look, the jury was still out. The recent paper by Cao and Zhu [**35**] sheds new light on this question. However, the opinions expressed so far are quite positive, and it looks very much like Perelman has finally affirmatively settled this amazingly difficult and influential conjecture. In the context of decidability questions, Perelman's work promises to have many important applications.

If Perelman is correct, this leads naturally to a very straightforward effective procedure for determining if a closed, three-dimensional, C^1-manifold M is a 3-sphere: First show that the fundamental group is trivial, which can be accomplished algorithmically by computing the edge-path group of a triangulation of M [**163**]. Using the same triangulation, it follows from Theorem 5.1 that the integral homology of M is effectively computable. Then if one computes that $H_0(M, \mathbb{Z}) = H_3(M, \mathbb{Z}) = \mathbb{Z}$, and $H_1(M, \mathbb{Z}) = H_2(M, \mathbb{Z}) = 0$, it follows that M is diffeomorphic, and *a fortiori* homeomorphic with S^3.

However, there already is an effective procedure, the Rubinstein–Thompson algorithm [**169**], for deciding within exponential time if a manifold is homeomorphic with S^3. This, of course, begs the question embodied in our next problem.

Question 5.9. *Develop an efficient algorithm based on the computation of the edge-path group and the integral homology as described above for deciding whether a closed manifold is homeomorphic with S^3. Then compare the computational complexity of this new algorithm with that of the Rubinstein–Thompson algorithm.* 1120?

Actually, Perelman's results claim to prove Thurston's Elliptization Conjecture for 3-manifolds (from which the Poincaré Conjecture follows immediately), which implies that all closed, connected, simply-connected, three-dimensional manifolds can be classified up to homeomorphism type. It appears that the elements of this classification theorem can be computed algorithmically, although this promises to be a daunting task owing to the techniques employed, not least of which are those generated by Hamilton's Ricci flow approach.

Question 5.10. *Within $\mathbb{R}^4$, prove that every closed, connected, simply-connected, three-dimensional C^1-submanifold is* TOP*-decidable and find estimates for the computational complexity of any relevant algorithms for deciding the homeomorphism types.* 1121?

Compact submanifolds of higher-dimensional Euclidean spaces: It follows from the Whitney Embedding Theorem [**149**] that every closed, four-dimensional C^1-manifold M can be embedded in $\mathbb{R}^N$ with $N \geq 9$. Four-manifolds provide some of the most intriguing and elegant TOP-decidable examples available, and they also yield important insights into the limitations of topological decidability. It follows from the work of Freedman, Donaldson, and others (as in [**56**, **79**]) that all closed, simply-connected, orientable, four-dimensional, C^1-manifolds M can be classified up to homeomorphism type. As a corollary, one obtains a proof of the Generalized Poincaré Conjecture for 4-spheres; namely, every simply-connected, homology 4-sphere is homeomorphic with the 4-sphere S^4.

One of the most beautiful aspects of this classification theory is the particularly simple criteria for determining the homeomorphism type, which comes out of the following observations. Elementary algebraic topology, Poincaré duality and the universal coefficient theorem for homology imply that $H_0(M, \mathbb{Z}) = H_4(M, \mathbb{Z}) = \mathbb{Z}$, $H_1(M, \mathbb{Z}) = H_3(M, \mathbb{Z}) = 0$, and $H_2(M, \mathbb{Z})$ is a free abelian group. This leads one to at least predict the important role in classification of 4-manifolds played by the bilinear, unimodular *intersection form* $\omega\colon H_2(M, \mathbb{Z}) \times H_2(M, \mathbb{Z}) \to \mathbb{Z}$. The classification theorem essentially states that the closed, oriented, simply-connected, differentiable four-dimensional manifolds are completely classified by their intersection forms. Consequently, we readily infer from Theorem 5.1 that these manifolds are also TOP-decidable. However, this result has, as far as we know, not appeared in the literature.

1122? **Question 5.11.** *Within Euclidean space* $\mathbb{R}^N$*, prove that all closed, orientable, simply-connected, four-dimensional* C^1*-submanifolds are* TOP*-decidable and estimate the computational complexity of the classifying algorithms.*

So 4-manifolds can lead to what may be considered to be among the best of times when it comes to topological decidability, but they also show us the worst of times—undecidability. It can be shown using simple manifold surgery techniques that every finitely presented group G (possibly very far removed from the trivial group) can be realized as the fundamental group of a closed, four-dimensional C^∞-manifold. Using this fact, and certain undecidability results for the isomorphism problem for groups, Markov proved that there exist certain 4-manifolds that are not TOP-decidable [**122**, **163**]. There are limits to the topological decidability of manifolds after all, and one need not look higher than four dimensions to find them. Naturally, this leads to several open problems that we leave to the reader to pose.

As higher dimensions provide more room for the techniques of differential topology to perform their mathematical magic, it is not surprising that the Generalized Poincaré Conjecture and the classification of closed, simply-connected, differentiable manifolds were proven by Smale [**162**], Stallings [**164**], Zeeman [**182**], and others more than a decade before Freedman's remarkable work. The earlier breakthroughs of Smale, Stallings and Zeeman employed a variety of differential topological techniques such as Morse theory, cobordism theory, and obstruction theory, all of which appear to be accessible to algorithmic formulations for manifolds in CDT_N^k and so we leave this subsection by posing the following (formidable) open problem.

1123? **Question 5.12.** *Prove that every closed, simply-connected, n-dimensional manifold in* CDT_N^k*, where* $k \geq 1$ *and* $n \geq 5$*, is* TOP*-decidable and estimate the computational complexity of any relevant classifying algorithms. In particular, consider the case of simply-connected, homology n-spheres.*

5.4.2. *Decidability of compact nonmanifolds.* Each of the decidability problems delineated for compact submanifolds in CDT_N^k have analogs—which are even more challenging—for compact varieties V that are not submanifolds. Taking our cue from the triviality of the decidability problems for manifolds embedded in Euclidean spaces of dimensions less than or equal to three, and expecting Thom–Mather theory (see [**19**, **26**, **27**, **84**, **123**, **168**, **178**] to reduce much of the work to submanifold strata in Equation 5.1 for which our previous observations provide much insight into decidability, we pose the following.

1124? **Question 5.13.** *Prove that every connected, compact subvariety V in* CDT_N^k *with* $k \geq 1$ *is* TOP*-decidable. Find tight upper bounds for the computational complexity of the resulting algorithms.*

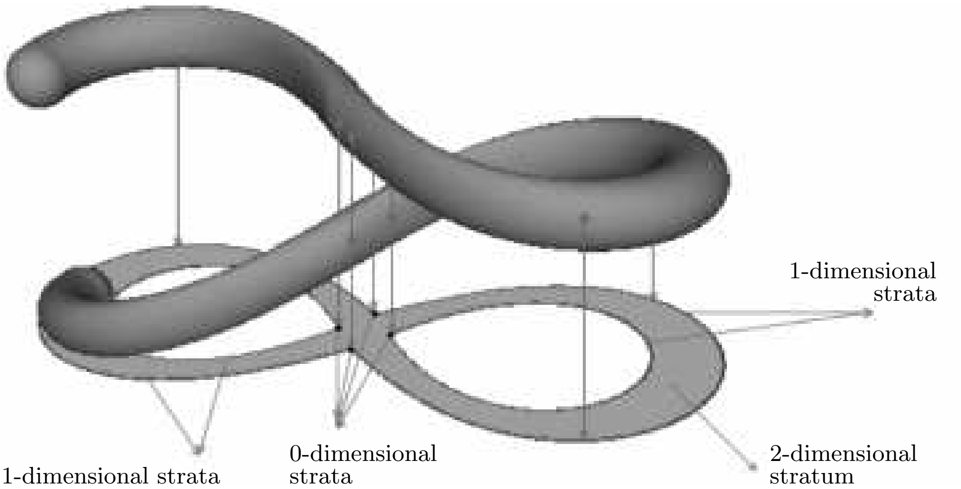


FIGURE 11. Manifold and projected sweep-like variety

It may be possible to show that the result in this theorem can be obtained in all higher dimensions as well, but clearly this would require some further restrictions on the homotopy type. Simple-connectedness might work, but this would severely restrict the types of nonmanifolds and many of the excluded ones would be apt to arise in a variety of applications. For, example, consider a thickened figure eight curve embedded in an Euclidean space of dimension four or higher. Another direction that one can pursue is to consider nonmanifolds obtained in a simple fashion from a compact manifold that is TOP-decidable. It is precisely this tack that we briefly follow in the remainder of this section, focusing upon compact sets that can be defined in terms of sweep-like operations.

Based upon extensive research on swept volumes [**1–3**, **24**, **25**, **27**, **28**, **175**], we are motivated make the following definition of a class of varieties that may yield to algorithmic classification.

Definition 5.8. A compact subvariety V of $\mathbb{R}^N$ is a sweep-like subvariety if there exists a compact submanifold M of $\mathbb{R}^{N+1} = \mathbb{R}^N \times \mathbb{R}$ such that $\Pi(M) = V$, where Π is the standard projection of $\mathbb{R}^N \times \mathbb{R}$ onto $\mathbb{R}^N = \mathbb{R}^N \times 0$, in which case V is said to be the projection of M.

A sweep-like variety is illustrated in Figure 11. Referring to this figure, we see that the self-intersection cell in the projection of the manifold has the appearance of an obstruction to lifting the variety to its regular preimage manifold of which it is the projection. This suggests that we can use a triangulation of the variety to identify this cell, in the manner of obstruction theory [**174**], in an algorithmic way. Thus, if the projecting manifold itself is topologically decidable, it appears that the same should be true of its image, which suggests that the following problem is solvable.

1125? **Question 5.14.** *Prove that every connected, compact, sweep-like subvariety V of $\mathbb{R}^N$ that is the projection of a compact, TOP-decidable, C^1-submanifold M of $\mathbb{R}^{N+1}$ is also TOP-decidable Find tight upper bounds for the computational complexity of any resulting algorithms.*

6. Computational topology resolution

A common practical concern is the development of algorithms that can produce appropriate topological representations upon models whose boundaries are formed by geometric intersections [**159**]. This is often known as 'topology resolution' and it affords many opportunities for additional research. The circumstances motivating this role for topological resolution have already been discussed in Section 3, particularly in Subsections 3.1 and 3.2.2.

One recent approach to managing the ill-formation of regular sets in computation [**146**, **147**] utilizes tubular neighborhoods [**94**], but presents a very broad definition of a family of sets, each based upon an initial set. An overview is that each incomplete boundary is used to develop a new family of candidate sets by building offsets of each boundary element. New Boolean operations are then defined upon this family of sets. One of the authors conjectures that there is a relationship to the Čech topology.

This work provides a point-set topological characterization for a family of sets such that each member closely approximates the original set according to a precise criterion, where it is clear that the family has some similarities to sets defined via interval arithmetic. The methods presented are appealing and will work for simple cases. However, as the geometry becomes more complex it remains of interest to understand a general approach to formulate these tubular neighborhoods, along with guarantees upon the properties of the family of sets generated and operators used within that family. In order to obtain such a family from a specific instantiated boundary model, it becomes essential to understand which conditions must be satisfied by the approximants, where an argument is given for homotopy equivalence [**157**].

1126? **Question 6.1.** *Is there a characterization of those tubular neighborhoods which can be used to define useful families of regular closed sets as alternatives to ill-formed computational representations?*

1127? **Question 6.2.** *Does the construction provide some meaningful relation to the Čech topology?*

6.1. Integration with numerical analysis.

6.1. Integration with numerical analysis. Another approach relies upon more classical techniques from numerical analysis, specifically the Whitney Extension Theorem [**177**], as captured in a recent doctoral thesis of M. Zidini [**183**] and several related preprints [**17**, **18**, **167**].

The strategy presented is to take the ill-formed geometry and use the Whitney Extension Theorem to extrapolate the imperfectly fitting boundary elements until a satisfactory manifold boundary is created. The emphasis is to build a theoretical model, not necessarily one that would be instantiated in any specific computational

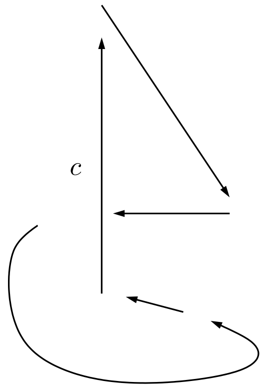


FIGURE 12. Gap analysis for Whitney extensions

representation. The intent would be to use this idealized model as a basis for developing rigorous error bounds as to how far any specific instantiation differed from this ideal. Some metrics are proposed for those measurements. Any two surface patches within the object boundary are said to be *adjacent* if they meet in a common boundary. Extensions of adjacent patches are proposed to ensure that they meet in a well-formed shared boundary, as a means to compensate for the numerical intersector errors previously discussed between surface patches.

The thesis makes the explicit assumption that "non-adjacent perturbed patches are disjoint." While this is standard, its use within computation raises the more subtle issue of the magnitude of the separation between these non-adjacent patches and the separation between adjacent patches.

Consider a finite set of surface patches whose union forms an object boundary, where the boundary curves for these surface patches have been created via the Grandine–Klein intersector [**86**] with an error preprocessor [**133**], so that the 'gaps' between adjacent surfaces were guaranteed to be no greater than λ in model space. One would hope that λ was chosen judiciously. Further, let δ denote the minimum distance between any two non-adjacent pathces. Generally, one would hope that $\lambda \ll \delta$. But, suppose, to the contrary, that $\lambda \gg \delta$.

To create a well-formed model from these surfaces, it would be appropriate to use the proposed Whitney extensions that would have perturbations on the order of λ, *but* perturbations of that magnitude could possibly introduce unwanted intersections between non-adjacent entities, as described below relative to the example of Figure 12. (Figure 12 uses curves for simplicity of exposition, but it should be clear that the example could easily be generalized to represent surfaces.[6] The curves should join to form a closed loop. This ideal connectivity is indicated by each arrowhead pointing to the end point of next segment.)

[6] For instance, each of the curves could serve as the spine of a swept surface having a generating curve of a circle of fixed radius.

For Figure 12, curve c lies in a plane perpendicular to the page, considered to be the plane $z = 0$. The image shown of c as a vertical line segment is merely its planar projection into the plane $z = 0$. The remaining curves shown all lie in the plane $z = 0$.

The other salient aspects of Figure 12 are summarized, as follows:

- The minimum distance δ is between c and the horizontal segment. This value of δ occurs at an interior point of c and at the left end point of the horizontal segment.
- The 'gap' between the left end point of the horizontal segment and its following free-form curve is the maximal value of λ.

Under these circumstances, the Whitney extensions to join properly the vertical segment and the free-form curve could create unintended intersections with c. If these gaps were the result of a construction process, such as a Boolean operation relying upon a numerical surface intersector, then a reasonable response might be to re-execute the procedures that generated the model, with tighter tolerances upon the numerical intersector so that one would have $\lambda \ll \delta$.

1128? **Question 6.3.** *Is it possible to provide practical criteria for the choice of λ, the separation distance between non-adjacent patches relative to δ, and the separation distance between adjacent patches?*

1129? **Question 6.4.** *If the errors resulting in these models being ill-formed as regular closed sets arose from some geometric construction process, such as a Boolean operation relying upon a numerical surface intersector, is a reasonable response to re-execute the procedures that generated the model, with tighter tolerances upon the numerical intersector so that one would have $\lambda \ll \delta$?*

1130? **Question 6.5.** *Are there implications that geometry should move from representation by specific instantiations into models that are more descriptive?*

1131? **Question 6.6.** *Might appropriate topological abstractions be more helpful than specific geometric coordinate based information?*

A very recent manuscript [**157**] argues that interpreting the inconsistencies between the geometric data and its connective information should rely upon a homotopy equivalence between the represented geometry and the intended exact set. The homotopy equivalent geometric sets are described as lying within the same tolerance zone. Additionally, graph theoretic and cell complex techniques are used to express and understand additional constraints that should be imposed upon these homotopies. In particular, it is proposed that these tolerance zones must be contractible for all cells that are homeomorphic to finite-dimensional Euclidean balls. Some further relationships are proposed to describe these homotopies in terms of the nerve of a collection of closed sets.

Question 6.7. *If homotopy equivalence is considered as a necessary condition for tolerant representations of geometry, what further conditions result in sufficiency?*

Question 6.8. *Can a practical algortihm be created to implement this theory regarding the homotopy equivalence between geometric representation and intent?*

6.2. The role of exact arithmetic. The study of 'exact arithmetic' arose from the computational geometry community [**181**] in recognition that many geometric predicates were critical to evaluation along boundaries. The question of being 'on' a boundary was equated to resolving whether numeric expressions were exactly equal to zero. In some cases, this can be done quite nicely. Assuming that all the input geometry coordinates are expressed exactly as rational numbers, then it is well known that roots for polynomials can be found within the field of algebraic numbers. So, one of the important aspects of exact arithmetic is to augment the typical floating point representation with additional data structures for radicals over the rationals.

Current language implementations for exact arithmetic have specific predicates for algebraic numbers [**142**]. So, solutions to $x^2 - 2$ can be represented exactly by these predicates. Then, these expression can be approximated to any number of bits specified by the user. Furthermore, algebraic operations are represented as directed acyclic graphics (DAG), with floating point values at the leaf nodes and algebraic operations at other nodes. In this sense, they are similar to CSG trees of Section 3.2.1. Since this DAG is the primary data structure, solutions can be adapted to user specified precision by just putting better approximations into the leaf nodes and being careful about error accumulation at the other nodes. There is a performance penalty for exact arithmetic. Efficient implementations are available for low-degree polynomial representations.

Question 6.9. *Can exact arithmetic be augmented to include non-algebraic numeric representations?* 1132?

Question 6.10. *What happens when the assumption of exact rational input is not met?* 1133?

The use of exact arithmetic can be contrasted with more classical techniques from numerical analysis. Specifically, the recent publication [**99**] presents a role for backward error analysis, with a reply included from proponents of exact arithmetic. This leads naturally to the next question.

Question 6.11. *What is the role of methods from numerical analysis, specifically backward error analysis, when there is uncertainty in the input data?* 1134?

Question 6.12. *Can exact arithmetic have competitive performance with approximate floating point geometric algorithms over high-degree polynomial representations?*

7. Computational topology and surface reconstruction

A significant catalyst for computational topology has been the problem of constructing an approximating surface mesh given only a sample of points from the surface. This problem was formalized and brought to the attention of the computer graphics community in a seminal 1992 paper [**97**]. Amenta and Bern [**9**, **10**] described the *crust* algorithm for which they could show, under some conditions on the surface and the sample, that the output approximates, geometrically, the surface from which the samples were drawn. A later simplification [**11**] of this algorithm was shown to produce a PL (triangulated) manifold homeomorphic to the surface from which the samples were taken, using a somewhat complicated argument involving covering spaces. These results have been extended to prove isotopy equivalence, with the following being a representative theorem [**13**].

Theorem 7.1. *Let F be a compact, C^2 2-manifold without boundary. Let S be a set of sample points of F such that for each $x \in F$, there exists a point $s \in S$ such that $d(x, s) < k\,\mathrm{LFS}(x)$, where $k = 0.085$ and $\mathrm{LFS}(x)$ is the minimum distance between x and the medial axis of F. Then, there is an algorithm that will take S as input and produce a PL approximation of F that is ambient isotopic to F.*

1135? **Question 7.1.** *What are necessary and sufficient conditions on a low-dimensional manifold to permit an ambient isotopic approximant as the manifold reconstruction?*

1136? **Question 7.2.** *What criteria are necessary and sufficient on the density of the sampling set on a low-dimensional manifold to permit an ambient isotopic approximant as the manifold reconstruction?*

1137? **Question 7.3.** *What is the appropriate topological equivalence relation to consider for manifold reconstruction?*

1138? **Question 7.4.** *Specifically, for manifolds without boundary, what are necessary and sufficient conditions on the normal field on the boundary to permit an ambient isotopic approximant as the manifold reconstruction?*

Recent work that may be helpful references in considering these questions include [**4**, **42**, **43**, **50**, **51**, **54**, **78**, **127**, **152**], with recent theorems appearing for the cases with boundary [**4**].

8. Computational topology and low-dimensional manifolds

Many of the 1-manifolds and 2-manifolds for geometric computing are described as spline functions [**87**, **141**]. These splines are typically defined over very simple domains, such as $[0, 1]$ and $[0, 1]^2$. While low-dimensional manifolds have their own subdiscipline within topology, it is consistent here to consider these manifolds in relation to generalized spline functions.

8.1. Background. The basic approach we outline here uses two steps to construct a function. In the first step, we model the *domain* of the function as an abstract manifold (this manifold need not have geometry associated with it). In the second step we define an embedding or immersion of the domain, e.g., to produce a surface. This second step is done piecemeal by defining local embedding or immersion functions on subsets of the domain, then blending the results using a partition of unity.

More formally, given a manifold M, a method for defining charts $\alpha_c(M) \to c \subset \mathbb{R}^n$ on M, immersion $E_c\colon c \to \mathbb{R}^m$ and blend $B_c\colon c \to \mathbb{R}$ functions for each chart, we can define a function on the entire manifold as follows:

$$E(p) = \frac{\sum_c B_c(\alpha_c)E_c(\alpha_c(p))}{\sum_c B_c(\alpha_c)} \tag{8.1}$$

To ensure this equation is valid, we place some constraints on the chart α_c and blend B_c functions. First, the charts must cover the manifold, i.e., they are a finite atlas. Second, the blend functions are non-zero over c. This ensures that the denominator is not zero. (Note: There's nothing that prevents the support of B_c being smaller than c, but it makes it harder to prove that the denominator is non-zero.) The E_c functions can be any function of continuity C^k over the region c (the continuity outside of c does not matter).

The continuity of the above equation is the continuity of its constituent parts. Therefore, to have a C^k function the α_c, E_c, and B_c functions must be at least C^k. The blend functions must also have their value and first k derivatives go to zero near the boundary of c. This ensures continuity at the boundaries of each chart.

For surfaces, the manifolds that make sense are planes, spheres, and hyperbolic disks tiled with $4n$-sided polygons (with edge pairs identified). The latter is one possible domain for n-holed (genus n) surfaces. This domain simplifies to the tiled plane for a standard (1-holed) torus. The E_c functions are typically polynomials or spline functions.

For reinforcement learning, the manifold is a combination of all possible actions and sensor readings, and the E_c function is a number that says how good it is to take that action with those sensor readings (essentially, a height field).

In animation, the manifold depends on the movement. Suppose a character is throwing a ball. A manifold that describes this motion (in a simplistic way) consists of a periodic value (where in the throw they are) and a release point (x, y, z). The function on the manifold is a set of joint angles for every joint in the body.

8.2. Problem statement. The problem can be loosely stated as follows. There exist some number of samples d_i of what the surface or function should look like; those samples may contain noise. Additionally, the parameter values p_i for the samples (i.e., where they are on the manifold) may also be known. The goal is to minimize $\sum_i \|d_i - E(p_i)\|$ where the p_i are given or they give the closest point to d_i on E, $\min_{p_i} \|d_i - E(p_i)\|$.

If the goal is interpolation of the points d_i then the sum should be zero.

In addition to the above approximation constraints, there is usually some form of "smoothness" constraint to guide what happens between the sample points. This can take several forms. One option is to minimize some combination of the second derivatives, such as the bending energy. A second option is to bound how much the surface varies from, e.g., a linear approximation to the data points.

A related set of constraints concerns "features" in the data, such as sharp edges and corners. In this case, it may be desirable for the function E to correctly model the edge or corner, i.e., to exhibit a discontinuity in differentiability.

8.3. Solving the problem. There are two stages to solving the problem. The first is to decide the chart placement (the α_c), the second is to fit the individual functions E_c. Ideally, the E_c functions agree where they overlap, i.e., for all charts c_i overlapping a point p, $E_{c_i}(\alpha_{c_i}(p))$ evaluates to the same thing. In this case, the shape of the blend function doesn't matter. In practice, the shape of the blend function has little effect on the final shape, so we can simply define the same blend function shape for all charts.

There are two options for solving for the free parameters in the E_c functions. The first is to fit each E_c locally to an appropriate subset of the data. The second is to fit all of the E_c simultaneously. The latter is, in general, more computationally expensive, but has the potential to produce better results.

Some observations:

- The more the charts overlap, i.e., the more non-zero terms in Equation 8.1, the smoother the result tends to be, but this increases the computational expense. "Smoother" is not a well-defined term here; clearly, the surface has the same *continuity* regardless of the overlap. However, there is an averaging effect that reduces the effect of local variation in the individual E_c.
- The size of the chart and the corresponding required complexity of the E_c function are inversely related. As the chart size decreases, the variation in E that E_c is responsible for decreases. In the limit, with an infinite number of charts we could use piecewise constant functions for the E_c.
- Given a fixed number of degrees of freedom for E_c the desired local variation in E also determines the size of the chart. In large, flat areas, we can use a single chart, but in regions with more variation we need more charts.
- Features such as sharp edges, can be modeled using a function E_c that is capable of representing a discontinuity. In this case, all of the other E_c functions need to be "masked out" or they may unduly influence that area. However, it may be difficult to use a single chart for a feature that spans most of the manifold.

The following are important open questions:

1139? **Question 8.1.** *What are the optimum size, shape, and amount of overlap for the charts? The answer to this question depends both on the data and on the choice of E_c. Optimum is a measure both of the fit (including a definition of smoothness) and computational tractability.*

1140? **Question 8.2.** *Beyond questions of charts for a known manifold, there is also the question of figuring out what the underlying manifold is for a given set of data points. The assumption is that the data points d_i arise from samples of a low-dimensional manifold embedded in a high-dimensional space. This is the field of* manifold learning *in computer vision; most of the techniques (principal components analysis, isomap, simple linear embedding) currently work only for planar manifolds, or largely convex (geometrically) spherical or cylindrical data* *sets. What unifying theory is possible for determining the appropriate underlying manifold for a given set of data points?*

9. Skeletal structures

Many of the previously discussed approaches to surface reconstruction in Section 7 use the *medial axis*, which, under specific hypotheses, can be shown to be a deformation retract [**179**]. This is an important concept, but its reliable and efficient computation poses many theoretical [**46**–**48**] and practical [**45**] challenges.

Between any two points, $x, y \in \mathbb{R}^3$, let $d(x, y)$ denote the usual Euclidean distance and for any two sets $X, Y \subset \mathbb{R}^3$, let $d(X, Y) = \inf\{d(x, y) : x \in X, y \in Y\}$.

Definition 9.1. Let $x \in \mathbb{R}^n$ and $S \subset \mathbb{R}^n$. A point $s \in S$ is a *nearest point on* S to x if $d(x, s) = \inf\{d(x, t) : t \in S\}$. The *medial axis* of S, is the closure of the set of all points that have at least two distinct nearest points on S.

This concept was originally defined for object recognition in the life sciences [**29**, **30**]. One investigation of the mathematical properties of the medial axis and its associated transform function [**44**] is restricted to geometry within the plane. More generally, there has been broad attention to the medial axis in $\mathbb{R}^n$ within the computer science literature, where the topological and differentiable investigations [**160**, **179**, **180**] are directly relevant to surface reconstruction work.

Both classical and contemporary research have emphasized the principle that many analytic attributes of surfaces can be determined using singularity theory and stratification theory [**1**–**3**, **27**, **28**, **41**, **46**–**48**, **55**, **144**, **158**, **168**, **178**]. In particular, singularities can be shown to correspond to possible self-intersections or non-manifold points and can be organized in Thom–Boardman form [**19**, **84**, **123**]. However, computational solutions for the associated nonlinear equations can be prohibitively expensive using many variants of Newton's method. Furthermore, other relevant exponential algorithmic bounds [**41**] appear to pose daunting computational difficulties. Recent singularity publications do offer promising techniques that could lead to efficient algorithmic preservation of ambient isotopy type [**145**, **149**, **163**, **174**], particularly in conjunction with recent findings by Blackmore [**27**, **28**] of approximate methods. The "skin surfaces" introduced in the context of biological modeling [**58**] have been shown to have isotopic approximating meshes [**118**]. The authors of this last paper note that their algorithms presume that the geometric input set is fixed, but this raises a question about about whether a given output would be appropriate for other the input sets.

Question 9.1. *Once a mesh is created, does it remain valid for some deformations of the input set, if those deformations are suitably constrained?* 1141?

The cut locus is similar to the medial axis and has been used in computational explorations of shape [**180**]. In particular, Wolter proves, for a rich class of surfaces, that C^2 continuity is not required to establish a positive distance between the surface and its cut locus, with a related corollary showing desirable smoothness properties of offsets of these surfaces.

Recognizing both the difficulties of approximating the medial axis and the sensitivity of the medial axis to small (though possibly inconsequential) changes in form, there has been recent mathematical work in proposing alternatives to the medial axis [**46**–**48**]. This work seeks the determination of relations between the skeletal structure proposed and the boundary of the original object, so that small changes in one will result in small changes in the other, where these alternative skeletal structures are often more topologically complex than the medial axis. The first of these publications [**46**] defines various geometric tools in support of these skeletal structures, and some of these tools hold promise for computational topology research, even as we note the distinction that the primary application of these skeletal generalizations has been to computer vision [**143**], as opposed to various simulation contexts.

The expected theory is likely to have some similarities to the use of the nerve simplicial complex technique previously invoked by Edelsbrunner and Shah [**70**]. There also appear to be similarities to the skeletal structures defined in the already cited papers by Damon and his coauthors [**46**–**48**, **143**], for their consideration of robust variants of the medial axis and applications in computer vision. The envelope may also be regarded as one of the level sets generated by the normal flow, so there may be opportunities to leverage the extensive classical and contemporary literature on level sets. Similarly, the extensive existing literature on the Minkowski sum, deserves careful study for a variety of applications.

1142? **Question 9.2.** *What are the appropriate skeletal structures and algorithms to extract critical topological information while reducing the representation?*

10. Computational topology and Biology on simplicial complexes

Topology studies global properties of geometric objects, like the number of connected components, tunnels, or cavities. The work on computational topology led by Edelsbrunner has had many interesting applications to biology [**60**]. His more theoretically fundamental work on Delaunay triangulations [**57**, **59**] is integral to these biological applications. This discussion presents those topics together. The Delaunay triangulations are typically classified as computational geometry, but the definition of their basic cells has a strong topological element. The triangulation is formed as a dual of a Voronoi diagram, which lies within a metric space, Z, having a metric $d\colon Z \to \mathbb{R}$. The Voronoi diagram presumes the existence of a finite set of points $Q = \{q_0, q_1, \ldots, q_n\}$ from Z. The Voronoi diagram is a collection of closed neighbhorhoods of the q_i, each containing one of the q_i. For each q_i its neighborhood is defined as the set of all $p \in Z$ such that $d(p, q_i) < d(p, q_j)$ for all $j \neq i$. Another related construct is that of α-shapes [**69**, **150**].

An overview article has appeared [**60**]. The techniques are based largely on simplicial complexes, computing invariants such as Euler characteristics, Betti-numbers and writhing numbers [**8**]. Additionally, Morse theory is invoked [**61**] to develop novel data representations for visualization algorithms. These ideas were the subject of a *New Directions* short course at the Institute for Mathematics and Applications [**63**]. One outcome was to relate computational Morse theory to

Forman's discrete Morse theory. Some of the contributions to the literature along these themes appear in various venues [**7**, **20**, **34**, **52**, **53**, **58**, **61**, **62**, **64**–**69**, **71**, **72**]. However, even this list is only partially representative of the broad and deep impact this research has had within the computational topology community.

Some of the techniques evolve more from algebraic topology methods, which has become an independently rich area in computational topology under the leadership of Edelsbrunner, as well as that of Carlsson [**36**]. The latter endeavors have are also integrated with statistics, forming a very rich subject area, which can only be mentioned here for the benefit of the interested reader. Two conferences on *Algebraic Topological Methods in Computer Science* have been held.

A summarizing question becomes

Question 10.1. *What role can discrete Morse theory play for the theoretical basis for algorithms in computational topology?* 1143?

Additional work on simplicial complexes emphasizes recovering topological invariants of a space from a finite set of noisy samples, parameterized within a high-dimensional Euclidean space. In order to have robustness versus undersampling and noise, a multiscale view of the space is created that contains information at all granularities. A space is constructed incrementally using a geometric criterion, obtaining a family of spaces. The spaces are not independent, but are related by inclusion maps that induce maps between the topological attributes in the spaces. The theory of *persistent homology* captures these relationships as lifetimes for the evolving attributes [**68**]. These lifetimes translate into a measure of importance for topology. So, persistence is a robust mechanism for recovering topology as it separates topological noise from features.

The traditional approach is to approximate the space by placing small balls around the samples and characterizing the combinatorics of the ball set. The resulting complex is simple but very expensive to compute. Unfortunately, no effective techniques are known for computing small complexes for points in high-dimensional spaces.

Question 10.2. *Can local methods be used to take advantage of the geometry to yield small complexes that would be computationally tractable?* 1144?

Often, one can generate a multiple-parameter family of spaces that describes a point set. For example, one might wish to track the topology of isosurfaces of both pressure and temperature of a jet flow across time. Recent progress in persistent homology indicates that a simple description is not possible for multiple parameters [**185**]. There is need for an approximation theory that allows access to the topological information contained in such a family.

Question 10.3. *Can robust invariants be computed for these multiparameter spaces?* 1145?

This summary represents recent issues posed largely from the joint work of Zomorodian and Carlsson and earlier work of Zomordian with Edelsbrunner.

There is much emphasis upon homological invariants, which lies beyond the articulated scope of this article. Nonetheless these aspects are included here because of their nascent state, portending that there may remain unresolved issues about the underlying topological spaces as this work matures further.

11. Finite approximation and (non-Hausdorff) topology

It has been known for almost 80 years that every compact Hausdorff space is the subspace of closed points of an inverse limit of finite T_0-spaces, and that finite T_0-spaces are essentially finite posets. For many years this seemed an oddity; why would anyone approximate the best known and best understood topological spaces by spaces that were simultaneously trivial and nonintuitive?

11.1. Adapted inverse limit approximation by T_0-spaces. The development of computing and its need for information in bits, and more particularly the work on digital topology from a purely topological viewpoint led to much more intuition on these finite T_0-spaces. As a result, Kopperman and Wilson proved that these inverse systems can be assumed to have very special maps, which they called *calming maps*. If this is done, the following traditional knowledge can be recast, as stated, below.

11.2. Topological invariants. The association between an abstract simplicial complex, which can be seen as a finite T_0-space, and its polytope in a finite-dimensional Euclidean space can be used as follows: The topological spaces that most often occur in science and engineering are the metric continua. These are often viewed as inverse limits of polyhedra and simplicial maps. The work by Kopperman and Wilson [**113**] has shown that these inverse systems of polyhedra and simplicial maps can be replaced by inverse systems of abstract simplicial complexes and calming maps in such a way that the inverse limit of the former is exactly the subspace of closed points of the inverse limit of the latter. Rather than the Euclidean polytopes and simplicial maps, which are determined by vertices and subject to round-off error, one can use precisely given finite posets and special order-preserving maps, also precisely given. Here are some issues that arise before these methods can be applied: While it has been known for about three years that the above approximation can be done, no algorithm for finding these finite posets and calming maps has been described and this method has not been used to approximate spaces. But the digital topology needed to understand the finite spaces was learned over a dozen years ago, in part by Kopperman and co-workers [**101**, **102**, **106**, **108**, **111**, **117**].

11.3. Topological consistency. Much of the relationship between this approximation method and basic general topology has been resolved. For example, if the finite T_0-spaces are connected, then so is their limit [**114**, **116**], and so is this subspace of closed points. Also, the relationship between the separation axioms (particularly complete regularity, normality and hereditary normality) and properties of the finite spaces and maps has been determined [**115**, **116**]. The authors are now preparing for publication results on replacing maps between the original spaces with maps between inverse systems of finite approximants of these original spaces. These results yield characterizations of the Stone–Čech and Wallman compactification in terms of such finite approximations (some of this was noted earlier [**76**]). But much more knowledge is needed about such replacement and its use in computation and the preservation of invariants of algebraic topology.

A primary view from domain theory is that many important computational topology properties correspond to open sets and not to specific Euclidean points or scalar values [**6**, **82**, **83**]. More specifically, Kopperman and his collaborators have characterized those topological spaces that are computable in the sense of domain theory [**109**]. A special case involves those that are inverse limits of polyhedra, creating an opportunity to include domain theoretic results into computational topology investigations.

A summarizing research question becomes:

Question 11.1. *What are the essential topological relations for visualization* and *how can reliance upon domain theory and these approximating systems improve upon the state-of-the-art to preserve key embedding (homotopy and homology) invariants of the models and spaces as they become visualized, both statically and dynamically?* 1146?

12. Algorithmic topology and computational topology

The work of creating KnotPlot [**154**, **155**] has been described as "topological drawing". By programs based upon Gaussian energy functionals, KnotPlot animates the process of unknotting and knot simplification on *specific examples* of knots. A key criterion is that the class of the knot is known *a priori*. This is an important aspect, as it is known [**91**] that the elementary problem of recognition of the piecewise linear unknot is in NP. Practical algorithms for knot recognition have proven elusive, but the problem remains an important stimulus for theoretical research [**89**].

This theoretical result provided valuable guidance to the work mentioned in Section 4 on isotopic approximations. Namely, it directed attention to just *preserving* the isotopy class of the original object *even when that classification was not known.* This is an example of the "beneficial symbiosis" anticipated [**63**] with algorithmic topology [**124**]. It leads to whether similar benefit can be gained by consideration of other algorithmic topology recognition problems, such as these summarized here.

The 3-sphere recognition problem starts with a given triangulation T and attempts to answer whether the underlying space $|T|$ is homeomorphic to the 3-sphere. It is shown that this problem lies in NP [**156**].

Question 12.1. *Is the 3-sphere recognition problem NP-hard?* 1147?

13. Computational topology workshop of 1999

To the best of our knowledge, the first broad workshop on Computational Topology was held in June, 1999 [**21**]. Its purpose was *direction finding* and its majority attendance was by scientists who are primarily recognized as computer scientists, though some pure mathematicians did attend and many of the participants are interdisciplinary in their work.

The report of this workshop is highly recommended for its broad coverage. Its impetus from computational geometers is reflected in the very applied nature of many of the topics and problems described. The report was not merely a description of technical problems, but also an attempt to identify areas, build community and develop an agenda for future research. As such, its purposes were somewhat different from the present article. Furthermore, because of the large number (22) of contributing coauthors, the report covers many subjects that are not intimately related to point-set topology. However, many of its findings are relevant for setting context. Some are quoted here. Furthermore, some specific problems do relate directly to this topology community, broadly considered. For instance, the definition of neighborhoods for differing topologies is a common problem of interest to many in the point-set topology community. Some problems, quoted below, mention the definition and representation of neighborhoods. Within the mathematics community, the specialty of low-dimensional topology is often viewed as being quite separate from that of point-set topology. However, problems from low-dimensional computational topology are presented here, because they depend upon such fundamental topological notions that it is hard to separate the fields. It is hoped that this blending of the subjects within computational topology might lead to more interaction among these communities within more established mathematical communities, hopefully to the benefit of mathematics at large. Those have been abstracted and updates provided, where relevant. Noticeably, several of these topics and problems are well-integrated with problems already posed within this article and that integration has been previously mentioned and is also noted, below.

The report begins with an emphasis upon the role of geometric computing to support the simulation of phsyical objects—"on scales that vary from the atomic to the astronomical."—emphasizing the role of topology in "Modeling the shapes of these objects and the space surrounding them." The role of information visualization is expressed as relying upon "shapes and motions" with obvious topological implications. The emphasis is upon support for geometric computing in that,

> "Some of the most difficult and least understood issues in geometric computing involve topology. Up until now, work on topological issues has been scattered among a number of fields, and its level of mathematical sophistication has been rather uneven. This report argues that a conscious focus on computational topology will accelerate progress in geometric computing."

While this specific focus on the benefits to geometric computing are understandable, this present article presents the point of view that topologists can make significant contributions to many aspects of computing. The "scattered" distribution throughout the literature is evident in the bibliography for this article, with cited publications appearing in mathematical and computer science venues, as well as within many different fields of engineering.

That the use of classical topology can "accelerate progress" has already been quite well documented in the literature. One notable success story has been the theorems generated by digital topologists. The wide-spread application, within the image processing community, of the Jordan curve and surface theorems to identify boundaries in images and partition viewed objects into parts lying inside and outside of those boundaries led to a contemporary study of these classical theorem, providing new proofs to apply to spaces that did not have T_2 separation properties [**110**]. These configurations of pixels on a computer screen were named *digital spaces*. While this seemed like an obvious use of a classical theorem, various unexpected subtleties occurred in algorithms in which this theory was applied. While it was unlikely that these difficulties would provide counterexamples that would invalidate such well-established theorems, it took the perspective of topologists to realize that

- the proofs of the classical Jordan separation theorems relied upon an assumption that the underlying topology was T_2,
- the digital spaces were discrete when modeled as T_2 topologies, and
- that weaker topologies were more descriptive of digital spaces.

These topologists then proved that the classical T_2 assumption was not needed and developed non-T_2 topologies for digital spaces. The rigorous consideration of these applied image problems led to extensions of classical theory and improved algorithms.

Some summarizing perspectives from this report state that "Topology separates global shape properties from local geometric attributes and provides a precise language for discussing these properties" and that "Mathematical abstraction can also unify similar concepts from different fields." These notions are, of course, well known to topologists, but it is of interest to understand that these aspects are now seen to be attractive in furthering the development of algorithms in robotics, molecular docking and geometric computing in general.

Some broad questions resulting from this report are summarized below, followed by more detailed sections with specific questions under each broader item. Again, the emphasis is upon topological issues that are most closely related to point-set topology, ignoring others that may have more of an algebraic topology or combinatorial topology emphasis.

13.1. Summary of broad questions.

Question 13.1. *How should shape be represented?*

Question 13.2. *How can topology preservation be ensured in converting from one shape representation to another?*

Question 13.3. *How can physical measurements, with sampling error and noise, be algorithmically converted into topologically valid shape representations, particularly for physical simulations that rely upon meshed geometry?*

Question 13.4. *How can "the development of algorithmic tools implementing topological concepts" [**63**] and "algorithmic questions in topology" be integrated for the benefit of both fields?'*

Because of preceding material, the key questions for each are tersely summarized.

13.2. Shape representation. This is consistent with the earlier remarks (Sections 2, 3 and 6) about the role of regular closed sets in solid modeling. Since this has already been discussed at some length, the relevant problems will be tersely stated, below.

Question 13.5. *Current shape representations include unstructured collections of polygons (with no specific connectivity information among geometric entities—often dubbed as 'polygon soup'), "polyhedral models, subdivision surfaces, spline surfaces, implicit surfaces, skin surfaces, alpha shapes", solid models, procedural models, digital and voxel models. What are the unifying topological constructs and how should they be expressed and implemented for efficient and robust algorithms?*

13.3. Topologically correct shape conversion. These issues have been discussed in some depth in Section 4 on approximation.

Question 13.6. *While there exist some methods for converting from one type of shape representation to another, these are mostly for polyhedral models and they are not totally rigorous or robust. How can topological principals be included in these shape conversions to both provide broad theory and improved algorithms? (We note that Section 4 has already discussed the inclusion of isotopy equivalence as a criterion for approximations (often PL ones) of smooth shapes in conjuction with traditional criteria of error bounds on the distance between one shape and its approximant.)*

1148? **Question 13.7.** *While classical topology has relied upon homeomorphisms for its primary equivalence relation, the geometric models in computing appear to need a stronger equivalence relation that includes correctness of the embedding within some low-dimensional topological (usually Euclidean) space. Is isotopy the preferred equivalence or is there need for even stronger equivalences such as diffeotopy?*

13.4. Shape acquisition algorithms and measurement error. Some of the dominant approaches here have avoided the issues of measurement error and noise. Recent abstractions [**152**] have proven theorems that leave open the opportunity to consider sample points with bounded measurement errors on a par with those that are exact samples. Several questions have already been articulated in previous sections.

Question 13.8. *How can these differing mathematical perspectives, across point-set and differentiable topology be best integrated for optimal shape-acquisition algorithms?*

13.5. Shape smoothness criteria. These issues range from unexpected appearance of non-smoothness in engineering design models to the need to represent non-smoothness in animation figures.

Question 13.9. *In some cases singularities arise because of numerical approxi-* 1149?
*mations made, which are inherent to a finite word length for numeric representations. In other cases, particularly for the motion picture industry, there are needs to model sharp changes in differentiability [***49***]. Some promising techniques have been presented that allow flexibility in moving gracefully between these needs [***165***]. Is there an appropriate topological abstraction that can be mapped easily to abstract data types that will permit appropriate representations of smoothness for differing applications?*

14. Conclusion

The bibliography is indicative of the breadth of interest in this subject, even though many references do not necessarily include the terminology "computational topology". As with any article presenting open problems, this one necessarily is reflective of the tastes and interests of the coauthors, where Sections 10, 12 and 13 are terse. This is not reflective of their scientific importance or impact, but rather an attempt to appeal to the expected point-set topology readership of this volume. In particular, the material presented here in those sections was directed towards emphasizing their general topology content, while showing their broader connections to other branches of topology for readers who might be interested in further consideration of these related subjects.

15. Acknowledgements

The authors wish to thank Herbert Edelsbrunner for some constructive criticism and insightful comments—most of which were made after lectures and during informal conversations at the recent Workshop on Computational Topology held on July 14, 2005 at Denison University, Granville. Thanks are also due to Ralph Kopperman and Yuriy Mileyko for numerous helpful conversations.

The following colleagues are acknowledged for their significant contributions to the indicated sections: C. Grimm (Section 8), R.D. Kopperman (Section 11) and V. Shapiro and N.F. Stewart (Section 6).

References

[1] K. Abdel-Malek, W. Seaman, and D. Blackmore, *On swept volume formulations: implicit surfaces*, Computer-Aided Design **33** (2001), 113–121.

[2] K. Abdel-Malek, W. Seaman, and H. J. Yeh, *NC verification of up to 5-axis machining processes using manifold stratification*, ASME J. Manufact. Sci. Eng. **122** (2000), 121–135.

[3] K. Abdel-Malek and H. J. Yeh, *Geometric representation of swept volume using Jacobian rank-deficiency conditions*, Computer-Aided Design **29** (1997), 457–468.

[4] K. Abe, J. Bisceglio, D. R. Ferguson, T. J. Peters, A. C. Russell, and T. Sakkalis, *Computational topology for isotopic surface reconstruction*, Theoret. Comput. Sci. (2006), Article in press.

[5] K. Abe, J. Bisceglio, T. J. Peters, A. C. Russell, D. R. Ferguson, and T. Sakkalis, *Computational topology for reconstruction of surfaces with boundary: integrating experiments and theory*, International Conference on Shape Modeling and Applications (M. Spagnuolo, A. Pasko, and A. Belyaev, eds.), IEEE Computer Society, 2005, pp. 290–299.

[6] S. Abramsky and A. Jung, *Domain theory*, Handbook of logic in computer science, Vol. 3, Oxford University Press, New York, 1994, pp. 1–168.

[7] P. K. Agarwal, H. Edelsbrunner, J. Harer, and Y. Wang, *Extreme elevation on a 2-manifold*, SCG'04: Proc. 20th Symposium on Computational Geometry, ACM Press, New York, 2004, pp. 357–365.

[8] P. K. Agarwal, H. Edelsbrunner, and Y. Wang, *Computing the writhing number of a polygonal knot*, Discrete Comput. Geom. **32** (2004), no. 1, 37–53.

[9] N. Amenta and M. Bern, *Surface reconstruction by Voronoi filtering*, Discrete Comput. Geom. **22** (1999), no. 4, 481–504.

[10] N. Amenta, M. W. Bern, and M. Kamvysselis, *A new Voronoi-based surface reconstruction algorithm*, ACM SIGGRAPH (1998), 415–421.

[11] N. Amenta, S. Choi, T. K. Dey, and N. Leekha, *A simple algorithm for homeomorphic surface reconstruction*, Proc. 16th Symposium on Computational Geometry, ACM Press, New York, 2000, pp. 213–222.

[12] N. Amenta, S. Choi, and R. K. Kolluri, *The power crust, unions of balls, and the medial axis transform*, Comput. Geom. **19** (2001), no. 2–3, 127–153.

[13] N. Amenta, T. J. Peters, and A. C. Russell, *Computational topology: ambient isotopic approximation of 2-manifolds*, Theoret. Comput. Sci. **305** (2003), 3–15.

[14] L.-E. Andersson, S. M. Dorney, T. J. Peters, and N. F. Stewart, *Polyhedral perturbations that preserve topological form*, Comput. Aided Geom. Design **12** (1995), no. 8, 785–799.

[15] L.-E. Andersson, T. J. Peters, and N. F. Stewart, *Selfintersection of composite curves and surfaces*, Comput. Aided Geom. Design **15** (1998), no. 5, 507–527.

[16] L.-E. Andersson, T. J. Peters, and N. F. Stewart, *Equivalence of topological form for curvilinear geometric objects*, Internat. J. Comput. Geom. Appl. **10** (2000), no. 6, 609–622.

[17] L.-E. Andersson, N. F. Stewart, and M. Zidani, *Error analysis for operations in solid modeling in the presence of uncertainty*, 2004, Preprint.

[18] L.-E. Andersson, N. F. Stewart, and M. Zidani, *Conditions for use of a non-selfintersection conjecture*, Comput. Aided Geom. Design (2006), To appear.

[19] V. I. Arnol′d, S. M. Guseĭn-Zade, and A. N. Varchenko, *Singularities of differentiable maps. Vol. I*, Birkhäuser, Boston, 1985.

[20] U. Axen and H. Edelsbrunner, *Auditory Morse analysis of triangulated manifolds*, Mathematical visualization (Berlin, 1997), Springer, Berlin, 1998, pp. 223–236.

[21] M. Bern et al, *Emerging challenges in computational topology*, 1999, Workshop Report. arXiv:cs.CG/9909001

[22] R H Bing, *The geometric topology of 3-manifolds*, American Mathematical Society, Providence, RI, 1983.

[23] J. Bisceglio, C. Mow, and T. J. Peters, *Boolean algebraic operands for engineering design*, 2006, Preprint.

[24] D. Blackmore, M. C. Leu, and Shih, *Analysis and modelling of deformed swept volumes*, Computer-Aided Design **26** (1994), 315–326.

[25] D. Blackmore, M. C. Leu, and L. P Wang, *The sweep-envelope differential equation algorithm and its application to NC machining verification*, Computer-Aided Design **29** (1997), 629–637.

[26] D. Blackmore and Y. Mileyko, *Computational differential topology*, Appl. Gen. Topol., To appear.

[27] D. Blackmore, Y. Mileyko, M. C. Leu, W. C. Regli, and W. Sun, *Computational topology and swept volumes*, Geometric and algorithmic aspects of computer-aided design and manufacturing, American Mathematical Society, Providence, RI, 2005, pp. 53–78.

[28] D. Blackmore, R. Samulyak, and M.C. Leu, *A singularity theory approach to swept volumes*, Int. J. Shape Modeling **6** (2000), 105–129.

[29] H. Blum, *A transformation for extracting new descriptors of shape*, Models for the Perception of Speech and Visual Form (W. Wathen-Dunn, ed.), MIT Press, Cambridge, 1967, pp. 362–380.

[30] H. Blum, *Biological shape and visual science (Part I)*, J. Theoret. Biol. **38** (1973), 205–287.

[31] M. Boyer and N. F. Stewart, *Modeling spaces for toleranced objects*, Internat. J. of Robotics Research **10** (1991), no. 5, 570–582.

[32] M. Boyer and N. F. Stewart, *Imperfect-form tolerancing on manifold objects: a metric approach*, Internat. J. of Robotics Research **11** (1992), no. 5, 482–490.

[33] M. Braverman and S. Cook, *Computing over the reals: foundations for scientific computing*, Notices Amer. Math. Soc. **53** (2006), no. 3, 318–329.

[34] P.-T. Bremer, V. Pascucci, H. Edelsbrunner, and B Hamann, *A topological hierarchy for functions on triangulated surfaces*, IEEE Trans. Vis. Comput. Graphics **10** (2004), 385–396.

[35] H.-D. Cao and X.-P. Zhu, *A complete proof of the Poincaré and geometrization conjectures—an application of the Hamilton–Perelman Theory of the Ricci flow*, Asian J. Math. **10** (2006), 165–492.

[36] G. Carlsson, *TMSCSCS: topological methods in scientific computing, statistics and computer science*, `http://math.stanford.edu/comptop/`.

[37] S.-L. Chang and M. S. R. Rocchetti, *Rendering cubic curves and surfaces with integer adaptive forward differencing*, ACM SIGGRAPH (1989), 157–166.

[38] F. Chazal and D. Cohen-Steiner, *A condition for isotopic approximation*, Graphical Models and Image Processing **67** (2005), 390–404.

[39] F. Chazal and A. Lieutier, *Topology guaranteeing surface reconstruction and medial axis approximation from noisy data*, Tech. Report 429, Institut de Mathématiques de Bourgogne, 2005, `http://math.u-bourgogne.fr/topolog/prepub/ApproxNoisyData3.pdf`.

[40] F. Chazal, A. Lieutier, and J. Rossignac, *Projection-homeomorphic surfaces*, SPM'05: Proc. ACM Symposium on Solid and Physical Modeling, ACM Press, New York, 2005, pp. 9–14.

[41] B. Chazelle, H. Edelsbrunner, L. J. Guibas, and M. Sharir, *A singly-exponential stratification scheme for real semi-algebraic varieties and its applications*, Automata, languages and programming (Stresa, 1989), Lecture Notes in Comput. Sci., vol. 372, Springer, Berlin, 1989, pp. 179–193.

[42] S.-W. Cheng, T. K. Dey, and E. A. Ramos, *Manifold reconstruction from point samples*, SODA'05: Proc. 16th ACM-SIAM Symposium on Discrete Algorithms, SIAM, Philadelphia, 2005, pp. 1018–1027.

[43] S.-W. Cheng, T. K. Dey, E. A. Ramos, and T. Ray, *Sampling and meshing a surface with guaranteed topology and geometry*, SCG'04: Proc. 20th Symposium on Computational Geometry, ACM Press, New York, 2004, pp. 280–289.

[44] H. I. Choi, S. W. Choi, and H. P. Moon, *Mathematical theory of medial axis transform*, Pacific J. Math. **181** (1997), no. 1, 57–88.

[45] T. Culver, J. Keyser, and D. Manocha, *Exact computation of the medial axis of a polyhedron*, Comput. Aided Geom. Design **21** (2004), no. 1, 65–98.

[46] J. Damon, *Smoothness and geometry of boundaries associated to skeletal structures. I. Sufficient conditions for smoothness*, Ann. Inst. Fourier (Grenoble) **53** (2003), no. 6, 1941–1985.

[47] J. Damon, *Smoothness and geometry of boundaries associated to skeletal structures. II. Geometry in the Blum case*, Compositio Math. **140** (2004), no. 6, 1657–1674.

[48] J. Damon, *Determining the geometry of boundaries of objects from medial data*, Int. J. Comp. Vision **63** (2005), no. 1, 45–64.

[49] T. DeRose, M. Kass, and T. Truong, *Subdivision surfaces in character animation*, ACM SIGGRAPH (1998), 85–94.

[50] T. K. Dey, *Curve and surface reconstruction*, Handbook of discrete and computational geometry (J. E. Goodman and J. O'Rourke, eds.), CRC Press, 2004, Second edition, pp. 677–692.

[51] T. K. Dey, *Curve and surface reconstruction: algorithms with mathematical analysis*, Cambridge University Press, Cambridge, 2006.

[52] T. K. Dey, H. Edelsbrunner, and S. Guha, *Computational topology*, Advances in discrete and computational geometry (South Hadley, MA, 1996), Contemp. Math., vol. 223, American Mathematical Society, Providence, RI, 1999, pp. 109–143.

[53] T. K. Dey, H. Edelsbrunner, S. Guha, and D. V. Nekhayev, *Topology preserving edge contraction*, Publ. Inst. Math. (Beograd) (N.S.) **66(80)** (1999), 23–45.

[54] T. K. Dey and P. Kumar, *A simple provable algorithm for curve reconstruction*, SODA'99: Proc. 10th ACM-SIAM Symposium on Discrete Algorithms, SIAM, Philadelphia, 1999, pp. 893–894.

[55] P. Dimitrov, J. Damon, and K. Siddiqi, *Flux invariants for shape*, Computer Vision and Pattern Recognition, IEEE Computer Society, 2003, pp. 835–841.

[56] S. K. Donaldson, *Yang–Mills invariants of four-manifolds*, Geometry of low-dimensional manifolds, 1 (Durham, 1989), Cambridge University Press, Cambridge, 1990, pp. 5–40.

[57] H. Edelsbrunner, *Algorithms in combinatorial geometry*, Springer, Berlin, 1987.

[58] H. Edelsbrunner, *Deformable smooth surface design*, Discrete Comput. Geom. **21** (1999), no. 1, 87–115.

[59] H. Edelsbrunner, *Geometry and topology for mesh generation*, Cambridge University Press, Cambridge, 2001.

[60] H. Edelsbrunner, *Biological applications of computational topology*, Handbook of discrete and computational geometry (J. E. Goodman and J. O'Rourke, eds.), CRC Press, Boca Raton, FL, 2004, pp. 1395–1412.

[61] H. Edelsbrunner, P.-T. Bremer, V. Pascucci, and B. Hamann, *A multi-resolution data structure for two-dimensional Morse–Smale functions*, VIS'03: Proceedings of the 14th IEEE Visualization 2003, IEEE Computer Society, Washington, DC, 2003, pp. 139–146.

[62] H. Edelsbrunner, M. A. Facello, P. Fu, J. Qian, and D. V. Nekhayev, *Wrapping 3D scanning data*, Three-dimensional image capture and applications, Proc. SPIE, vol. 3313, SPIE Press, Bellingham, WA, 1998, pp. 148–158.

[63] H. Edelsbrunner and J. Harer, *IMA workshop, July 6–16*, 2004, `http://www.ima.umn.edu/new-directions/2004NDshort-course/`.

[64] H. Edelsbrunner and J. Harer, *Jacobi sets of multiple Morse functions*, Foundations of computational mathematics: Minneapolis, 2002, Cambridge University Press, Cambridge, 2004, pp. 37–57.

[65] H. Edelsbrunner, J. Harer, A. Mascarenhas, and Pascucci V., *Time-varying Reeb graphs for continuous space-time data*, SCG'04: Proc. 20th Symposium on Computational Geometry, ACM Press, New York, 2004, pp. 366–372.

[66] H. Edelsbrunner, J. Harer, V. Natarajan, and V. Pascucci, *Morse–Smale complexes for piecewise linear 3-manifolds*, SCG'03: Proc. 19th Symposium on Computational Geometry, ACM Press, New York, 2003, pp. 361–370.

[67] H. Edelsbrunner, J. Harer, and A. Zomorodian, *Hierarchical Morse–Smale complexes for piecewise linear 2-manifolds*, Discrete Comput. Geom. **30** (2003), no. 1, 87–107.

[68] H. Edelsbrunner, D. Letscher, and A. Zomorodian, *Topological persistence and simplification*, Discrete Comput. Geom. **28** (2002), no. 4, 511–533.

[69] H. Edelsbrunner and E. P. Mücke, *Three-dimensional alpha shapes*, ACM Transactions on Graphics **13** (1994), no. 1, 43–72.

[70] H. Edelsbrunner and N. R. Shah, *Triangulating topological spaces*, Internat. J. Comput. Geom. Appl. **7** (1997), no. 4, 365–378.

[71] H. Edelsbrunner and R. Waupotitsch, *A combinatorial approach to cartograms*, Comput. Geom. **7** (1997), no. 5–6, 343–360.

[72] H. Edelsbrunner and A. Zomorodian, *Computing linking numbers of a filtration*, Algorithms in bioinformatics (Århus, 2001), Lecture Notes in Comput. Sci., vol. 2149, Springer, Berlin, 2001, pp. 112–127.

[73] A. V. Evako, R. Kopperman, and Y. V. Mukhin, *Dimensional properties of graphs and digital spaces*, J. Math. Imaging Vision **6** (1996), no. 2–3, 109–119.

[74] R. Farouki, *Closing the gap between CAD model and downstream application*, SIAM News **32** (1999).

[75] D. R. Ferguson, *An industrial look at CAGD within Boeing*, Invited plenary presentation, Sixth SIAM conference on geometric design, 1999.

[76] J. Flachsmeyer, *Zur Spektralentwicklung topologischer Räume*, Math. Ann. **144** (1961), 253–274.

[77] J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes, *Computer graphics: principles and practice*, Addison Wesley, 1997, Second Edition in C.

[78] D. Freedman, *Combinatorial curve reconstruction in Hilbert spaces: a new sampling theory and an old result revisited*, Comput. Geom. **23** (2002), no. 2, 227–241.

[79] M. H. Freedman, *The topology of four-dimensional manifolds*, J. Differential Geom. **17** (1982), no. 3, 357–453.

[80] A. Gain and A. Dodgson, *Preventing self-intersection under free-form deformation*, IEEE Trans. Vis. Comput. Graphics **7** (2001), no. 4, 289–298.

[81] E. Garduño and G. T. Herman, *Implicit surface visualization of reconstructed biological molecules*, Theoret. Comput. Sci. **346** (2005), no. 2–3, 281–299.

[82] G. Gierz, K. H. Hofmann, K. Keimel, J. D. Lawson, M. W. Mislove, and D. S. Scott, *A compendium of continuous lattices*, Springer, Berlin, 1980.

[83] G. Gierz, K. H. Hofmann, K. Keimel, J. D. Lawson, M. W. Mislove, and D. S. Scott, *Continuous lattices and domains*, Cambridge University Press, Cambridge, 2003.

[84] M. Golubitsky and V. Guillemin, *Stable mappings and their singularities*, Springer, New York, 1973.

[85] T. A. Grandine, *Applications of contouring*, SIAM Rev. **42** (2000), no. 2, 297–316.

[86] T. A. Grandine and F. W. Klein, IV, *A new approach to the surface intersection problem*, Comput. Aided Geom. Design **14** (1997), no. 2, 111–134.

[87] C. M. Grimm, *Simple manifolds for surface modeling and parameterization*, SMI'02: Proceedings of the Shape Modeling International 2002, IEEE Computer Society, Washington, DC, 2002, p. 237.

[88] E. L. Gursoz, Y. Choi, and F. B. Prinz, *Vertex-based representation of non-manifold boundaries*, Geometric modeling for product engineering, North-Holland, 1990, pp. 107–130.

[89] W. Haken, *Theorie der Normalflächen*, Acta Math. **105** (1961), 245–375.

[90] R. Hartshorne, *Algebraic geometry*, Springer, New York, 1977.

[91] J. Hass, J. C. Lagarias, and N. Pippenger, *The computational complexity of knot and link problems*, J. ACM **46** (1999), no. 2, 185–211.

[92] G. Hemion, *The classification of knots and 3-dimensional spaces*, The Clarendon Press Oxford University Press, New York, 1992.

[93] G. Herman, *3D imaging in Medicine*, CRC Press, 2000, 2nd Edition.

[94] M. W. Hirsch, *Differential topology*, Springer, New York, 1976.

[95] J. G. Hocking and G. S. Young, *Topology*, Addison Wesley, 1961.

[96] C. M. Hoffmann, *Geometric & solid modeling*, Morgan Kaufmann, San Mateo, CA, 1989.

[97] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle, *Surface reconstruction from unorganized points*, ACM SIGGRAPH (1992), 71–78.

[98] A. Jackson, *Conjectures no more?: consensus forming on the proof of the Poincaré and geometrization conjectures*, Notices Amer. Math. Soc. **53** (2006), no. 8, 897–901.

[99] D. Jiang and N. F. Stewart, *Backward error analysis in computational geometry*, Sixth Annual International Workshop on Computational Geometry and Applications (CGA '06) (Glasgow, UK), Lecture Notes in Comput. Sci., vol. 3980, Springer, Berlin, 2006, pp. 50–60.

[100] T. Kaczynski, K. Mischaikow, and M. Mrozek, *Computational homology*, Springer, New York, 2004.

[101] E. Khalimsky, R. Kopperman, and P. R. Meyer, *Boundaries in digital planes*, J. Appl. Math. Stochastic Anal. **3** (1990), no. 1, 27–55.

[102] E. Khalimsky, R. Kopperman, and P. R. Meyer, *Computer graphics and connected topologies on finite ordered sets*, Topology Appl. **36** (1990), no. 1, 1–17.

[103] J. Kister, *Small isotopies in Euclidean spaces and 3-manifolds*, Bull. Amer. Math. Soc. **65** (1959), 371–373.

[104] D. E. Knuth, *The art of computer programming. Vol. 2*, Addison Wesley, 1981.

[105] T. Y. Kong and E. Khalimsky, *Polyhedral analogs of locally finite topological spaces*, General topology and applications (Middletown, CT, 1988), Lecture Notes in Pure and Appl. Math., vol. 123, Marcel Dekker Inc., New York, 1990, pp. 153–164.

[106] T. Y. Kong, R. Kopperman, and P. R. Meyer, *A topological approach to digital topology*, Amer. Math. Monthly **98** (1991), no. 10, 901–917.

[107] T. Y. Kong and R. D. Kopperman, *Digital topology*, Encyclopedia of general topology (K. P. Hart, J. Nagata, and J. E. Vaughan, eds.), Elsevier Science, 2004, pp. 428–432.

[108] R. Kopperman and Y. Kong, *Using general topology in image processing*, Geometrical problems of image processing (Georgenthal, 1991), Akademie-Verlag, Berlin, 1991, pp. 66–71.

[109] R. Kopperman, H.-P. A. Künzi, and P. Waszkiewicz, *Bounded complete models of topological spaces*, Topology Appl. **139** (2004), 285–297.

[110] R. Kopperman, P. R. Meyer, and R. G. Wilson, *A Jordan surface theorem for three-dimensional digital spaces*, Discrete Comput. Geom. **6** (1991), no. 2, 155–161.

[111] R. D. Kopperman, *The Khalimsky line as a foundation for digital topology*, Shape in Picture (O. Ying-Lie et al, eds.), Springer, New York, 1994, pp. 3–20.

[112] R. D. Kopperman, *Topological digital topology (invited contribution)*, Discrete geometry for computer imagery (A. Nyström, G. Sanniti di Baja, and S. Svensson, eds.), Lecture Notes in Comput. Sci., vol. 2886, Springer, New York, 2003, pp. 1–15.

[113] R. D. Kopperman, V. V. Tkachuk, and R. G. Wilson, *The approximation of compacta by finite T_0-spaces*, Quaest. Math. **26** (2003), no. 3, 355–370.

[114] R. D. Kopperman and R. G. Wilson, *Finite approximation of compact Hausdorff spaces*, Topology Proc. **22** (1997), Summer, 175–200.

[115] R. D. Kopperman and R. G. Wilson, *On the role of finite, hereditarily normal spaces and maps in the genesis of compact Hausdorff spaces*, Topology Appl. **135** (2004), 265–275.

[116] R. D. Kopperman and R. G. Wilson, *Separation and connectedness in spectral compactification*, Houston J. Math., To appear.

[117] V. Kovalevsky and R. Kopperman, *Some topology-based image processing algorithms*, Papers on general topology and applications (Flushing, NY, 1992), Ann. New York Acad. Sci., vol. 728, New York Acad. Sci., 1994, pp. 174–182.

[118] N. Kruithof and G. Vegter, *Approximation by skin surfaces*, SM'03: Proc. 8th ACM Symposium on Solid Modeling and Applications, ACM Press, New York, 2003, pp. 86–95.

[119] T. Maekawa, N. M. Patrikalakis, T. Sakkalis, and G. Yu, *Analysis and applications of pipe surfaces*, Comput. Aided Geom. Design **15** (1998), no. 5, 437–458.

[120] M. Mäntyalä, *Computational topology: a study on topological manipulations and interrogations in computer graphics and geometric modeling*, Finnish Academy of Technical Sciences, Helsinki, 1983.

[121] R. Marabini, C. O. S. Sorzano, S. Matej, J. J. Fernández, J. M. Carazo, and G. T. Herman, *3d reconstruction of 2d crystals in real space*, IEEE Trans. Image Processing **13** (2004), 546–561.

[122] W. S. Massey, *Algebraic topology: An introduction*, Harcourt, Brace & World, Inc., New York, 1967.

[123] J. N. Mather, *Stratifications and mappings*, Dynamical systems (Bahia, 1971), Academic Press, New York, 1973, pp. 195–232.

[124] S. Matveev, *Algorithmic topology and classification of 3-manifolds*, Springer, Berlin, 2003.

[125] J. Milnor, *Lectures on the h-cobordism theorem*, Princeton University Press, Princeton, NJ, 1965.

[126] J. Mitchell, *Topological obstructions to blending algorithms*, Comput. Aided Geom. Design **17** (2000), no. 7, 673–694.

[127] N. Mitra and A. Nguyen, *Estimating surface normals in noisy point cloud data*, SCG'03: Proc. 19th Symposium on Computational Geometry, ACM Press, New York, 2003, pp. 322–328.

[128] G. Monge, *Application de l'analys à la geometrie*, Bachelier, Paris, 1850.

[129] E. L. F. Moore, *Computational topology of spline curves for geometric and molecular approximations*, Ph.D. thesis, The University of Connecticut, 2006.

[130] E. L. F. Moore and T. J. Peters, *Computational topology for geometric design and molecular design*, Mathematics for Industry: Challenges and Frontiers: A Process View: Practice and Theory (D. R. Ferguson and T. J. Peters, eds.), SIAM, 2005, pp. 125–139.

[131] E. L. F. Moore, T. J. Peters, and J. A. Roulier, *Preserving computatational topology by subdivision of quadratic and cubic Bézier curves*, Computing, To appear.

[132] J. W. Morgan, *Recent progress on the Poincaré conjecture and the classification of 3-manifolds*, Bull. Amer. Math. Soc. (N.S.) **42** (2005), no. 1, 57–78.

[133] C. Mow, T. J. Peters, and N. F. Stewart, *Specifying useful error bounds for geometry tools: an intersector exemplar*, Comput. Aided Geom. Design **20** (2003), no. 5, 247–251.

[134] J. R. Munkres, *Elementary differential topology*, Princeton University Press, Princeton, NJ, 1966.

[135] J. R. Munkres, *Elements of algebraic topology*, Addison Wesley, 1984.

[136] N. M. Patrikalakis and T. Maekawa, *Shape interrogation for computer aided design and manufacturing*, Springer, Berlin, 2002.

[137] T. J. Peters, J. Bisceglio, D. R. Ferguson, C. M. Hoffmann, T. Maekawa, N. M. Patrikalakis, T. Sakkalis, and N. F. Stewart, *Computational topology for regular closed sets (within the I-TANGO project)*, Topology Atlas Inv. Contrib. **9** (2004), no. 1, 12 pp. arXiv:math.GN/0412557

[138] T. J. Peters and D. W. Rosen, *The role of topology in engineering design research*, Research in Engineering Design **8** (1996), no. 2, 81–98.

[139] T. J. Peters, D. W. Rosen, and S. M. Dorney, *The diversity of topological applications within computer-aided geometric design*, Papers on general topology and applications (Flushing, NY, 1992), Ann. New York Acad. Sci., vol. 728, New York Acad. Sci., 1994, pp. 198–209.

[140] T. J. Peters and N. F. Stewart, *Computational topology for CFD: theorems, criteria and issues*, Proc. First MIT Conference on Computational Fluid and Solid Mechanics, Elsevier Science, 2001, pp. 131–134.

[141] L. Piegl and W. Tiller, *The NURBS book*, Springer, New York, 1997.

[142] S. Pion and C. K. Yap, *Constructive root bound for k-ary rational input numbers*, SCG'03: Proc. 19th Symposium on Computational Geometry, ACM Press, New York, 2003, pp. 256–263.

[143] S. Pizer et al, *Deformable M-reps for 3D medical image segmentation*, Int. J. Comp. Vision **55** (2003), no. 2–3, 85–106.

[144] S. Pizer, K. Siddiqi, G. Székely, J. Damon, and S. Zucker, *Multiscale medial loci and their properties*, Int. J. Comp. Vision **55** (2003), 155–179.

[145] S. Plantinga and G. Vegter, *Contour generators of evolving implicit surfaces*, SM'03: Proc. 8th ACM Symposium on Solid Modeling and Applications, ACM Press, New York, 2003, pp. 23–32.

[146] J. Qi and V. Shapiro, *Epsilon-regular sets and intervals*, International Conference on Shape Modeling and Applications, IEEE Computer Society, 2005, pp. 310–319.

[147] J. Qi and V. Shapiro, *Epsilon-topological formulation of tolerant solid modeling*, Computer-Aided Design **38** (2006), 367–377.

[148] A. A. G. Requicha, *Representations for rigid solids: theory, methods and systems*, ACM Computing Surveys **46** (1999), 185–211.

[149] T. B. Rushing, *Topological embeddings*, Academic Press, New York, 1973.

[150] T. Sakkalis and C. Charitos, *Approximating curves via alpha shapes*, Graphical Models and Image Processing **61** (1999), 165–176.

[151] T. Sakkalis and T. J. Peters, *Ambient isotopic approximations for surface reconstruction and interval solids*, SM'03: Proc. 8th ACM Symposium on Solid Modeling and Applications, ACM Press, New York, 2003, pp. 176–184.

[152] T. Sakkalis, T. J. Peters, and J. Bisceglio, *Application of ambient isotopy to surface approximation and interval solids*, Computer-Aided Design **36** (2004), 1089–1100.

[153] T. Sakkalis, G. Shen, and N. M. Patrikalakis, *Topological and geometric properties of interval solid models*, Graphical Models and Image Processing **63** (2001), 163–175.

[154] R. Scharein, *The KnotPlot Site*, `http://www.knotplot.com/`.

[155] R. Scharein, *Interactive topological drawing*, Ph.D. thesis, The University of British Columbia, 1998.

[156] S. Schleimer, *Sphere recognition lies in NP*, 2004, Preprint. `arXiv:math.GT/0407047`

[157] V. Shapiro, *Homotopy conditions for tolerant geometric queries*, Preprint.

[158] V. Shapiro, *Maintenance of geometric representations through space decompositions*, Internat. J. Comput. Geom. Appl. **7** (1997), no. 1–2, 21–56.

[159] G. Shen, T. Sakkalis, and N. M. Patrikalakis, *Analysis of boundary representation model rectification*, SMA'01: Proc. 6th ACM Symposium on Solid Modeling and Applications, ACM Press, New York, 2001, pp. 149–158.

[160] E. C. Sherbrooke, N. M. Patrikalakis, and F-E. Wolter, *Differential and topological properties of medial axis transforms*, Graphical Models and Image Processing **58** (1996), 574–592.

[161] *Point clouds in imaging science*, SIAM News **37** (2004), no. 7, 3 pp.

[162] S. Smale, *Generalized Poincaré's conjecture in dimensions greater than four*, Ann. of Math. (2) **74** (1961), 391–406.

[163] E. H. Spanier, *Algebraic topology*, McGraw-Hill, New York, 1966.

[164] J. R. Stallings, *Polyhedral homotopy-spheres*, Bull. Amer. Math. Soc. **66** (1960), 485–488.

[165] I. F. Stewart and A. R. Foisy, *Arbitrary-degree subdivision with creases and attribute*, J. Graphical Tools **9** (2004), no. 4, 3–18.

[166] N. F. Stewart, *A sufficient condition for correct topological form in tolerance specification*, Computer-Aided Design **25** (1993), 39–48.

[167] N. F. Stewart and M. Zidani, *Transfinite interpolation for well-definition in error analysis in solid modelling*, 2006, Dagstuhl Seminar 06021, January 8–13.

[168] R. Thom, *Ensembles et morphismes stratifiés*, Bull. Amer. Math. Soc. **75** (1969), 240–284.

[169] A. Thompson, *Thin position and the recognition problem for S^3*, Math. Res. Lett. **1** (1994), no. 5, 613–630.

[170] R. B. Tilove and A. A. G. Requicha, *Closure of Boolean operations on geometric entities*, Computer-Aided Design **12** (1980), 219–220.

[171] A. Varshney, *Hierarchical geometric approximations*, Ph.D. thesis, University of North Carolina–Chapell Hill, Department of Computer Science, 1994.

[172] G. Vegter, *Computational topology*, Handbook of discrete and computational geometry, CRC Press, Boca Raton, FL, 1997, pp. 517–536.

[173] H. B. Voelcker and A. A. G. Requicha, *Geometric modelling of mechanical parts and processes*, IEEE Computer **10** (1977), no. 12, 48–57.

[174] C. T. C. Wall, *Surgery on compact manifolds*, American Mathematical Society, Providence, RI, 1999.

[175] G. W. Wang, J. Sun, and X. Hua, *The sweep-envelope differential equation algorithm for general deformed swept volumes*, Comput. Aided Geom. Design **17** (2000), no. 5, 399–418.

[176] K. J. Weiler, *Topological structures for geometric modeling*, Ph.D. thesis, Rensselaer Polytechnic Institute, 1986.

[177] H. Whitney, *Analytic extensions of differentiable functions defined in closed sets*, Trans. Amer. Math. Soc. **36** (1934), no. 1, 63–89.

[178] H. Whitney, *Tangents to an analytic variety*, Ann. of Math. (2) **81** (1965), 496–549.

[179] F.-E. Wolter, *Cut loci in bordered and unbordered Riemannian manifolds*, Doctoral dissertation, Technical University of Berlin, Department of Mathematics, 1985.

[180] F.-E. Wolter, *Cut locus and medial axis in global shape interrogation and representation*, Design Laboratory Memorandum 92-2, MIT, 1993.

[181] C. Yap and T. Dubé, *The exact computation paradigm*, Computing in Euclidean geometry, World Scientific Publishing, Singapore, 1995, pp. 452–492.

[182] E. Zeeman, *The Poincaré conjecture for $n \geq 5$*, Topology of 3-manifolds and related topics, Prentice Hall, Englewood Cliffs, NJ, 1962, pp. 198–204.

[183] M. Zidini, *Validité d'un modèle QuasiNURBS interpolant des données géométriques incertaines*, Doctoral dissertation, Université de Montréal, 2006.

[184] A. Zomorodian, *Topology for computing*, Cambridge University Press, Cambridge, 2005.

[185] A. Zomorodian and G. Carlsson, *Computing persistent homology*, Discrete Comput. Geom. **33** (2005), no. 2, 249–274.