Topological Regularity in Semantics of Geometric Models

Qi, J. [a] , Shapiro, V. [a,*] , and Stewart, N. F. [b]

[a] Spatial Automation Laboratory, University of Wisconsin-Madison
1513 University Ave, Madison, WI 53706-1572, USA
E-mails: qi@students.wisc.edu, vshapiro@engr.wisc.edu

[b] Département d'informatique et de recherche opérationnelle,
Université de Montréal, CP6128, Succursale CentreVille,
Montréal, Qc, H3C 3J7, Canada
E-mail: stewart@iro.umontreal.ca

August 16, 2005

Abstract. Two major approaches to defining the semantics of inaccurate boundary representations have been proposed in the literature. They are referred to here as single-set semantics, and class-of-sets semantics, respectively. The two approaches differ in the nature of the topological regularity on which they are based (classical regularity *vs.* ϵ-regularity). It is shown in this paper that the two models are complementary, and that each may be useful, depending on the context. As an illustrative example, an elementary proof is given for a point-membership-classification result that is sufficiently general to be of practical interest, but sufficiently special to be transparent.

Keywords: semantics, regularity, point-membership, inaccurate representations, broken models

* Corresponding author

1 Introduction

The first significant study of the problem of attaching formal meaning to computer representations, in the context of shape interrogation and geometric modeling, was contained in work by Requicha and Voelcker that began in the 1970s. It was observed there that first, we must distinguish between a representation and the subset (or the class of subsets) that the representation specifies, since there may be many different representations for the same subset, or the same class of subsets. But, further, these early papers introduced formal semantics related to several other relevant aspects of the problem:

- an ideal *solid* (an r-set [1], which is defined as a bounded, semi-analytic, regular-closed subset of R^3);
- a *class* of ideal solids [2, Sec. 3.5], (which may be defined [3] using set inclusion as $\mathcal{C} = \{S : S_- \subseteq S \subseteq S_+\}$, with various possible conditions on S, S_- and S_+); this is appropriate in cases where an individual solid is not specified (an example [3, Fig. 4] is shown in Figure 1);
- a certain method of representation, referred to by the name of *constructive solid geometry*, or CSG (which defines a solid procedurally, by means of a binary tree); and
- another method of representation, referred to by the name of *boundary representation* (which defines a solid in terms of a finite two-dimensional cell complex, embedded in R^3, with injective attaching maps and, possibly, multiply-connected faces [4]).

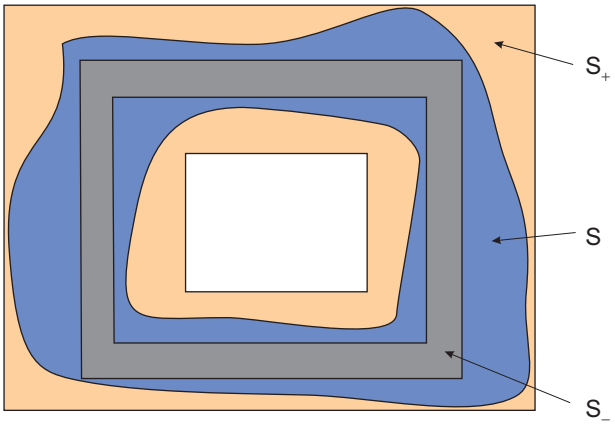

Figure 1: Specifying a class of solids

Other, different, representations and semantics were introduced later, by other authors, that are appropriate in certain circumstances. For example, the *selective geometric complexes* [5] provide a semantics for objects, along with a corresponding representation for these objects, in cases where for practical purposes the objects may be considered to be infinitely thin, or where objects are not composed of homogeneous material (see also [6]). In the present paper, however, we restrict our attention to the case of objects composed of homogeneous material.

For many practical reasons, the kernel of a modern commercial solid modeler is based on a boundary representation, and such representations (which use trimmed-NURBS patches[1]) suffer inherently from error. This error implies that they are inaccurate, and may contain many kinds of geometric and topological inconsistencies. The main sources of these errors are as follows: uncertainty in input data; the use of low finite-precision numerical representations for numbers which require higher precision, or which may, even in perfectly ordinary circumstances, be irrational; the use of finite-precision arithmetic to implement arithmetic operations; and, the use of low-degree polynomials to approximate high-degree intersection curves. Two common situations where these errors cause serious and costly difficulties are Boolean set operations, and

[1] NURBS (Non-Uniform Rational B-Splines).

translation between different kinds of boundary representations. These factors have led to various efforts to define the semantics of inaccurate boundary representations, as a first step towards proving theorems about the robustness of numerical algorithms applied to these representations. Such semantics supplies the meaning for, and must be based on, the data provided by a "user" – a generic term we will use throughout the paper to refer to a human user and/or computer application.

The present authors have, in separate papers [7, 8, 9], proposed two different approaches to defining the semantics of inaccurate boundary representations. One of the main differences between these two approaches is the nature of the topological regularity on which they are based. The two approaches will be described very briefly now, and in more detail in the next section.

The first approach was used in the context of trimmed-NURBS representations, and depended on the Whitney Extension Theorem to introduce Quasi-NURBS surfaces that provide well-defined solids [7]. The mathematical theory was based on the classical concept of *regularity*, and the underlying concepts of *interior* and *closure* in the topology of R^3. This approach introduced the concepts of *well-formedness* of an (inaccurate) representation, along with the definition of the (single) regular-closed set specified by such a representation, and showed how these concepts can be used to support a classical backward error analysis [7, 8]. Similar concepts can be introduced, by an appropriate interpretation of the "detail vectors" suggested in [10], for the case of subdivision-surface representation of non-manifold r-sets, a method which is becoming predominant in graphical-animation software.

The second approach [9] was based on a new concept of regularity, called ϵ-*regularity*, which captures the idea of tolerating small (ϵ-size) irregularities in the forms of errors, uncertainty, and/or approximation. We can view this approach as a generalization of the classical concept of regularity, and also as an extension of the idea of representing a class of solids in the form

$$\mathcal{C} = \{S : S_- \subseteq S \subseteq S_+\}, \tag{1}$$

as mentioned above (see [2], and [3, 11, 12]). Assigning a semantics to inaccurate and inconsistent representations using ϵ-regularity provides, in particular, an alternative way to deal with the gaps and overlaps inherent in trimmed-NURBS representations. It can also be used, however, in more general contexts, as will be described below.

The purpose of this paper is to show that the two semantic models, just described, are complementary, and that each may be useful, depending on the context. We will do this by describing circumstances where each is appropriate, and by proving some simple facts in each case, to illustrate the fundamental ideas.

The organization of the remainder of the paper is as follows. In Section 2 we compare the two models, discussing in particular the Quasi-NURBS semantics for boundary representations, and the Virtual-Interval representational method. In Section 3 we give an elementary proof for a point-membership-classification result, to illustrate clearly the contexts in which each of the semantic models is useful. Section 4 is the conclusion.

2 Two approaches to the semantics of inaccurate boundary representations

The Quasi-NURBS semantics for inaccurate data [7, 8] was introduced in the context of the standard trimmed-patch representation [13, 14]. It defines defines a single (unique) set to be interpreted as *the* set specified by a well-formed representation, and we will refer to it therefore as an example of "single-set semantics". The concept of regularity used in single-set semantics is classical regularity in the usual topology of R^3. In contrast to these ideas, certain representational methods related to the concept of ϵ-regularity can be viewed as associating a *class* of sets, or even *classes* of sets, with the inaccurate data (thus providing a "class-of-sets semantics"). The ϵ-regularity approach can be used in the contexts of standard trimmed-patch representations, of subdivision-surface representations, or more generally. In this section we will describe single-set and class-of-sets semantics, and discuss the tradeoffs between them.

2.1 Single-set semantics

Figure 2 shows a collection of geometric data, and the associated topological data, in a trimmed patch representation for an ordinary solid cube. The topological data, in this case, is simply the ordinary three-

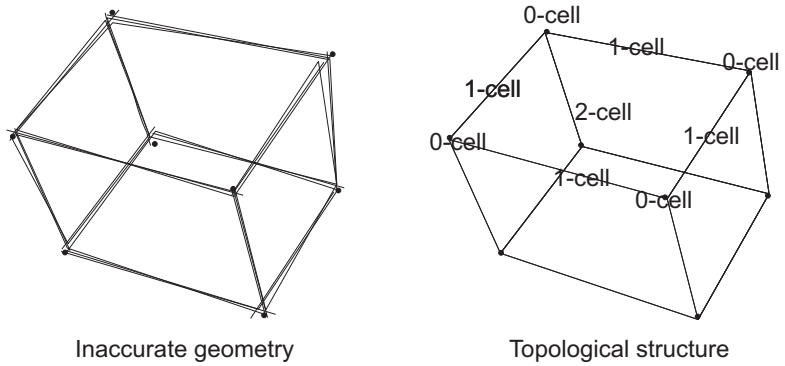

Figure 2: Geometric and topological data for an approximate cube

dimensional hypercube, with eight 0-cells, twelve 1-cells, and six 2-cells. There are six geometric faces, which correspond to the six 2-cells of the topological data. Similarly, there are are twelve explicit geometric boundary curves between the faces, corresponding to the twelve 1-cells of the topological data, and there are eight explicit geometric vertices, corresponding to the eight 0-cells of the topological data. As illustrated in Figure 2, none of the geometric entities necessarily fit together as they should: edges of two adjacent geometric faces may overlap, or be separated by a gap, or both; a geometric boundary curve may fail to coincide with its adjacent face edges; and, finally, the geometric vertices will usually not correspond to the corners of the adjacent geometric faces, nor even to the endpoints of the incident geometric boundary curves.

Let us suppose that, for a cube with the approximate dimensions shown in Figure 2, the problem of entering the solid into the modeling system is *well-conditioned*; that is, if there is uncertainty about the exact value that is intended for one of the geometric dimensions, perturbing this value does not have a noticeable effect on, say, the image displayed by a certain renderer. Similarly, suppose the algorithm to create this solid (for example, by extruding an approximate rectangle lying approximately in a plane) is *stable*. Such an algorithm will certainly incur small errors, due to the use of finite-precision arithmetic, but the assumption just mentioned means that these errors have no significant effect from the point of view of the user [8]. Even with these assumptions, and even for a solid as simple as a cube, the geometric data of the resulting model is mathematically inconsistent, as illustrated in Figure 2. If we wish to assign a meaning to this inconsistent data, in order to prove theorems about the the results of subsequent operations applied to the solid, how should we proceed?

One approach to this question is to assign (under certain weak hypotheses on the data which permit the assignment to be made), a single set S that is considered to be the unique set specified by the inconsistent data Δ. (We write $S \models \Delta$.) This set S is first *proved* to satisfy many desirable conditions. For example, it is proved that the geometric characteristics of S are very similar to those of the provided data Δ, that the boundary ∂S satisfies certain smoothness criteria, that under suitable conditions the topological properties of S are the same as those of the provided data (and consistent with the previously mentioned geometric data), and so on. Then, it is *assumed* that S is an appropriate interpretation of the inconsistent data Δ.

The outline just given is a brief description of the approach taken in [7, 8]. There, it was assumed that the given geometric data for a trimmed-NURBS patch is as shown in Figure 3: the domain D of a patch is a subset of $D_0 = [0,1]^2$ that is defined by so-called *p-curves* $\boldsymbol{p}$ within D_0; further, the underlying NURBS surface is defined by $\boldsymbol{F}$, and the trimmed patch itself by $\boldsymbol{F}[D]$; also, between two adjoining trimmed patches $\boldsymbol{F}[D]$ and $\boldsymbol{F}'[D']$, there is an explicit parametric boundary curve $\boldsymbol{b}(t)$, which approximates the corresponding edges of $\boldsymbol{F}[D]$ and $\boldsymbol{F}'[D']$; and, finally, at each corner formed by a group of adjoining patches, there is an explicit geometric vertex $\boldsymbol{v}$ which approximates the endpoints of $\boldsymbol{b}(t)$, and which approximates the corners of the adjoining patches. This geometric data, along with the topological data mentioned earlier, constitute the

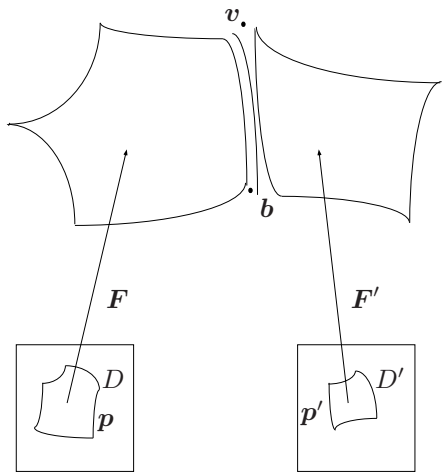

Figure 3: Two adjoining trimmed-NURBS patches

given data Δ. Then, the given geometric faces and boundary curves of the trimmed-NURBS representation were perturbed, or slightly adjusted, to yield what was called a Quasi-NURBS set. The adjusted faces of the Quasi-NURBS set S fit together exactly, in a way that is consistent with the given topological data. Furthermore, the size of the adjustment in a geometric face is no larger than the largest discrepancy already existing in the face at its boundary, according to the given data. Also, the rate of variation of the necessary adjustment, as measured by a Lipschitz constant over the entire domain D, is no larger than the rate of variation already present in the given face along its boundary. (Informally, we might say that the G^0 and G^1 quality of each face in the Quasi-NURBS set S is as consistent with the given data as it possibly can be.) Finally, smoothness guarantees[2] for the boundary of S were given.

It is important to be clear about the fundamental assumption involved here: as already stated, it is *assumed* that S is an appropriate interpretation of the inconsistent data Δ. In order to convince the end user that S is in fact an appropriate interpretation of Δ, we can list all of the desirable characteristics of the set S satisfying $S \models \Delta$, as in the preceding paragraph; we can, further, point out that a user who has provided inconsistent information, and who also views this data as a representation of an r-set, should not object if we take the data to represent some particular r-set that is nowhere in error by more than the error inherent in the original data; we can, also, observe that the *uncertainty* inherent in the user's data (measurement error, and rounding error incurred when entering the data into the computer) may well be as large or larger than the data adjustment involved in matching S with Δ; and, finally, we can observe that error must be tolerated in the eventual manufacture of the solid, and discrepancies in the provided data should be smaller than this. But ultimately, in order to proceed, we must *assume* that the end user accepts S as an appropriate interpretation of the inconsistent data Δ.

Suppose, however, that in view of the remarks above, this assumption is acceptable. We are then in the enviable position of having available a unique, well-defined mathematical object about which we can prove theorems. For example, we might prove that certain attributes, such as texture coordinates, can be correctly associated with the faces of the objects [16, 17]. Or, we might prove that a certain algorithm correctly computes the surface area of the boundary, or part of the boundary, to within a specified error. Or, we might prove that the straightforward algorithm to effect rigid motions on the solid produces another well-formed solid which is correct to within a specified error. And, similarly, we are as well-positioned as we possibly can be to prove that more complicated algorithms, such as set-intersection algorithms, are stable

[2]In [7], only continuity was assured, although the extension theorems necessary to obtain higher degrees of continuity are available [15].

when applied to given input data [7, 8].

2.2 Class-of-sets semantics

In contrast to the ideas of Section 2.1, the approach described in [9] is based on associating a *class* of sets, and often even *classes* of sets, with inaccurate data. A typical model or engineering artifact undergoes many transformations, translations, and other changes over the duration of its life cycle. The model's topology and geometry may be subject to any number of additional application-dependent constraints, including constraints on smoothness and engineering parameters (such as dimensions and tolerances). Under these conditions, a combinatorially equivalent well-formed set, such as a Quasi-NURBS set, or a set obtained by another single-set-semantics approach, may not exist, or may not be easily computable (even theoretically) for a particular inconsistent representation. More generally, data exchange standards do not specify the topology or the smoothness of the representations, and many geometric repair procedures modify both the geometry and topology of the representation. In such situations, the single-set semantics is not applicable. Even if a suitable well-formed representative does exist for any particular instance of the model, there is no guarantee that it will preserve all other constraints throughout the model's life cycle.

To establish the context for the subsequent discussion, we note that we are now concerned not with the use of a single boundary-defining representational method, such as trimmed-NURBS patches, but rather with a more general environment where many different representations may be involved, including representations that might not be known in advance, and even including representations whose nature might not be clearly understood by the end user. The intuitive idea of representing sets in terms of their boundaries is still present, but at this point little is assumed about the nature of these boundaries. For example, they might be defined by trimmed-NURBS patches; or, by data originating from trimmed-NURBS patches but which contains inconsistencies, as in Section 2.1; or, by a triangular mesh; or, say, by a triangular mesh badly in need of repair. In fact, in principle, they might be defined by a fractal, or some other complicated mathematical process that uses given numerical data. This minimalist view is particularly useful in situations, such as translation between different kinds of boundary representations, where it may be necessary to deal simultaneously with more than one interpretation of the given data. Additional geometric and topological properties of the represented sets will however need to be known, or have to be assumed, for development of specific algorithms, as will be explained in Section 2.2.3 and illustrated in Section 3.2.4.

2.2.1 The Virtual-Interval representational method

The class-of-sets semantics is assigned to a given geometric representation via an associated representational method, which we call the *Virtual-Interval* representational method. We begin our discussion, as in Section 2.1, by describing the mechanics of the representational method. The name "Virtual-Interval" arises from the fact that the representation is based on intervals of the form $\{S : S_- \subseteq S \subseteq S_+\}$, which is suggestive of the "virtual conditions" used in mechanical tolerancing [18].

We will again denote the given data (*i.e.*, a collection of numbers stored in the computer), for a specific set, by Δ, but it should be understood that initially nothing is assumed known about Δ: it might correspond to any of the cases listed in the second paragraph of Section 2.2, or to some representational method unknown to us.

We define the Point Membership Classification (PMC) function by

$$\Pi_\Delta : R^3 \mapsto \{in,\ out,\ on\};$$

this function defines the output of an actual implemented version of a procedure that attempts to compute whether a point in R^3 is, or is not, in the set S represented by Δ, or whether the point is on the boundary. We next define

$$S_{in} = \{\boldsymbol{x} : \Pi_\Delta(\boldsymbol{x}) = in\},$$

and

$$S_{out} = \{\boldsymbol{x} : \Pi_\Delta(\boldsymbol{x}) = out\},$$

from which it follows that

$$S_{in} \subseteq S_{out}^c,$$

where c denotes set complement. (Usually we will think of S_{in} as a bounded set, and S_{out} as an unbounded set, but nothing we have said so far implies that this is the case.)

Suppose now that we can prove that there exist sets S_- and S_+, depending on the choice of PMC algorithm, satisfying

$$S_- \subseteq S_{in} \subseteq S_{out}^c \subseteq S_+, \tag{2}$$

where S_- and S_+ are known to be in some sense close. Then the PMC algorithm, in conjunction with the theorem establishing (2), induces a class of sets

$$\{S : S_- \subseteq S \subseteq S_+\}$$

and we can think of the pair

$$[S_-, S_+]$$

as representing the set associated with Δ. This is the Virtual-Interval representational method. It can be seen that this approach involves a *class-of-sets* semantics, with S represented procedurally (*albeit* approximately) through the PMC function Π_Δ and its implementing algorithms.

An example illustrating how to prove the existence of such sets S_- and S_+ will be given below, in Section 3.2, but here we will discuss the approach by introducing the plausible hypothesis that the level of precision of the PMC algorithm is known, and that it can be used to define the sets S_- and S_+.

Without making further assumptions, there is nothing to exclude, say, a Π_Δ which produces *in* for an open rectangle, except for a certain number of holes (isolated points) near the middle of the rectangle; *in* for a certain number of isolated points outside and far from the rectangle; and, *out* for all other points (see Figure 4). But, in practice, users will generally have a good idea of the precision of the PMC algorithm. For

Figure 4: Black: *in*. White: *out*

example, with many CAD systems the user can set tolerances on entities such as faces and edges. Similarly, simple voxelization or octree-based methods, such as those used for model repair, can provide a reasonable estimate for a "maximum gap size" δ.

Suppose, then, that the geometric data in Δ is a collection of geometric entities (subsets of R^3, such as faces, edges, voxels, *etc.*, of a geometric representation Δ), and that the union of these entities is $G \subseteq R^3$. If it is believed that the maximum gap size for the PMC algorithm being used is δ, it may be reasonable to assume that

$$\bigcup_{\boldsymbol{x} \in G} B_\delta(\boldsymbol{x}), \tag{3}$$

where $B_\delta(\boldsymbol{x})$ is the open ball of diameter δ about x, determines regions S_- and S_+ such that (2) is satisfied, as illustrated in Figure 5. Furthermore, it may be possible to find easily computable subsets of these regions. Requiring such a hypothesis places increased responsibility on the user, but as we will illustrate in Section 3, below, the hypothesis will often be appropriate.

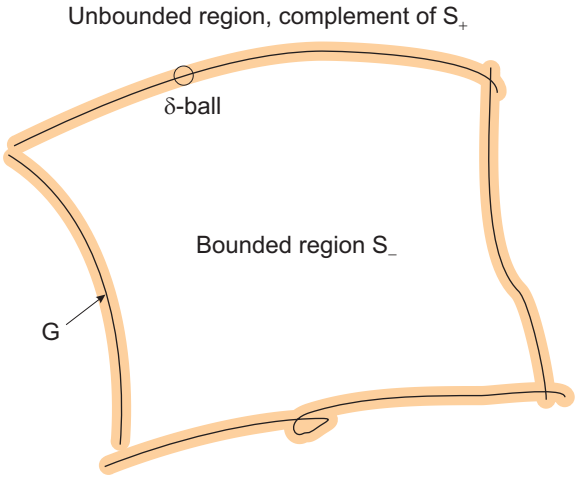

Figure 5: Covering G with δ-balls

2.2.2 The concept of ϵ-regularity

Interpreting a representation Δ as a class of sets in a virtual interval $[S_-, S_+]$ introduces a new complication. With a single-set semantics, the represented set is assumed or proved to be a valid r-set: a compact, regular and semi-analytic subset of R^3. This definition clearly does not apply to a virtual interval $[S_-, S_+]$. We could require that every set in the interval is an r-set, but this requirement would limit the usefulness of the Virtual-Interval method. For a detailed discussion of this and related issues, the reader is referred to [9], but here we simply mention the key properties of a virtual interval that make it useful in practice. A virtual interval should be "regular" in some sense similar to the classical notion of a regular-closed or regular-open set. But since we have no explicit notion of exact boundary, the new notion of regularity must *tolerate* some errors and imperfections of small size measured by a distance between S_- and S_+. In turn, this implies that S_- and S_+ should be in some sense close to each other.

In the context of the class-of-sets semantics, these requirements are expressed by means of an *ϵ-regular interval*. Indeed, in [9] it was argued that a suitable class-of-sets semantics is captured through the ϵ-topological operations k_ϵ, i_ϵ, and the notion of ϵ-regular interval requiring that

$$i_\epsilon(S_+) \subseteq S_- \qquad \text{and} \qquad k_\epsilon(S_-) \supseteq S_+, \tag{4}$$

where $\epsilon \geq 0$, and $S_- \subseteq S_+$. (For $\epsilon = 0$, these concepts reduce to their usual counterparts in the ordinary topology of R^3.) When $[S_-, S_+]$ is an ϵ-regular interval, the set of points in $S_+ \setminus S_-$ plays the role a "thickened boundary" that hides small imperfections and inconsistencies. This concept embodies what is perhaps the most striking difference, between the two major approaches to assigning semantics to inconsistent data, namely, that the class-of-sets semantics does not require preservation of the specific combinatorial structure of the model's boundary. Thus, the class-of-sets semantics does not guarantee the topological or even the geometric similarity of the sets in the class, at least not in the classical sense. On the other hand, if $[S_-, S_+]$ is an ϵ-regular interval, the Hausdorff distances between the sets S_-, S_+, their complements, and their boundaries are all less than ϵ [9].

2.2.3 Class-of-sets semantics in boundary representations

We return now to the motivation for the class-of-sets approach, in the context of boundary representation. In practice, the amount of inconsistency in a representation tends to be determined by the algorithms being applied to the representation. If the same underlying data Δ is to be used by a range of algorithms, with varying precision, its semantics will in practice be determined by PMC algorithms of different precision, corresponding, say, to the functions Π_Δ and Π'_Δ, which will therefore determine different classes of

sets $[S_{in}, S_{out}]$ and $[S'_{in}, S'_{out}]$. This situation arises often when a representation is used in a collaborative environment, or in the case of translation between systems with different precision.

We must be careful, however, not to confuse or confound the problem and the high-level method. To simplify the notation, consider a simple case where the problem input is defined by a single solid: for example, the problem of computing the intersection of the input set with a fixed hyperplane. Different end users, perhaps using different systems, would interpret the same underlying data Δ in different ways, as defined by the two different PMC functions Π_Δ and Π'_Δ; this is illustrated in Figure 6. They might also use

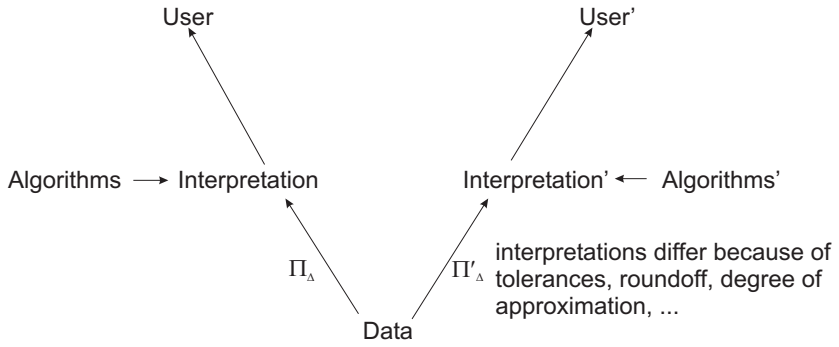

Figure 6: Multiple interpretations

different methods to compute the cross-section, and the precision of the method used by *User* may be related to the precision of Π_Δ (while the precision of the method used by *User'* is likely related to the precision of Π'_Δ). But the method and the PMC function are different things: Π_Δ is part of the definition of the problem to be solved. Thus, for example, *User* may be interested in the difference in the cost and accuracy of two different cross-section methods applied to the problem defined by Δ and Π_Δ.

Suppose now that Δ is a boundary representation, and the precision of Π_Δ is a constant δ that is measured by the maximum error in distance to the faces, edges, and vertices in Δ. In this case, the δ-cover defined in (3) plays the role of the thickened boundary representation, with all other points classifying either *in* or *out*, and thereby inducing the set interval $[S_-, S_+]$. However, in order to ensure that the PMC function Π_Δ makes sense, this interval must be ϵ-regular for some predefined value $\epsilon \geq \delta$. For example, if contrary to the assumption in Section 2.2.1, δ is smaller than the maximum size gap between faces, edges, and vertices, then $S_+ \setminus S_-$ may not be a connected set, raising the possibility that the boundary representation Δ does not bound any interior. Even if δ is larger than the maximum gap size, but $[S_-, S_+]$ is not ϵ-regular, the closeness of sets in the interval cannot be guaranteed. This implies that treating the δ-cover defined in (3) as a thickened boundary is in this case arbitrary and is not likely to be consistent with the intended semantics.

Furthermore, ϵ-regularity of the interval $[S_-, S_+]$ is a necessary but not sufficient condition for using a PMC algorithm with precision δ on the data in Δ, because ϵ-regularity does not say anything about the topological properties of the sets in the interval[3]. Indeed, there are no guarantees that the boundaries of S_- and S_+ are homeomorphic or in the same homotopy class. If such a guarantee is required, it must come from additional knowledge or from an assumption based on the origin and type of data in the boundary representation Δ. We will give an example of such an assumption in Section 3.2.4 in the context of a specific PMC algorithm.

3 Comparison of the semantic models

In this section it will be shown that the two semantic models, described above, are complementary, and that each is useful in the right context.

[3]This is consistent with the fact that the classical notion of regularity guarantees homogeneity of the solid's interior but does not say anything else about the topology of set [1].

3.1 General outline

A general outline, describing typical circumstances under which each of the two semantic models is appropriate, is as follows. If it is appropriate to make assumptions such as those of Section 2.1 where, for a NURBS representation, conditions were given permitting a proof that $S \models \Delta$, then for many applications the single-set semantics will be the most useful choice. On the other hand, if it is *not* appropriate to make the assumptions of Section 2.1, but it is appropriate to make some other assumption, such as an assumption implying that (3) determines sets satisfying (2), then the class-of-sets semantics is the most useful (and indeed the only) choice.

For example, consider the straightforward algorithm to transform S by a rigid motion, by applying the rigid motion to the control points of each of the trimmed patches $\boldsymbol{F}[D]$, to the control points of each explicit boundary curve $\boldsymbol{b}$, and to each explicit geometric vertex $\boldsymbol{v}$, using ordinary floating-point arithmetic. Let the *computed* representation of the transformed set be Δ_R, and let $R(S)$ be the theoretically exact result of applying the rigid motion R to S. It is easy to prove, using standard *a priori* bounds for floating-point arithmetic [19, Ch. 3], that under weak conditions the computed representation Δ_R is well-formed, and specifies a unique set S_{approx} such that $S_{approx} \models \Delta_R$, and the Hausdorff distance between $R(S)$ and S_{approx} is on the order of floating-point precision (about 10^{-7} for IEEE single precision [20]). In fact, it is possible to prove that the faces of the two Quasi-NURBS sets are close, on a pairwise basis. It would clearly be inappropriate to discard such strong conclusions, by using a class-of-sets semantics, if the hypotheses of the single-set semantics are justified.

As a second example, we might use numerical quadrature techniques, in the context of the single-set semantics, to estimate the area of a face of the represented set S. Rigorous proofs about such an operation would not be possible using the class-of-sets approach alone: it will be necessary to introduce additional hypotheses about analytic and/or topological properties of the geometric model, such as those used in the single-set theory. But this, of course, is exactly what should be done: there is nothing to prevent use of the single-set approach in conjunction with the class-of-sets approach.

It may also happen, however, that it is *not* appropriate to make the assumptions of Section 2.1. For example, suppose we are dealing with a representation Δ that suffers from ill-definition arising from the circumstances described in Section 2.2.3. If it is appropriate to make some other assumption, such as an assumption implying that (3) determines sets satisfying (2), then the class-of-sets semantics is the most useful (and indeed the only) choice. There is no possibility, in such a case, of using single-set semantics, or of obtaining results about the topological form of a computed approximation to a set transformed by a rigid motion. But this does not mean that nothing can be proved. Suppose that Δ represents a valid solid in a sense that the virtual interval $[S_-, S_+]$ is ϵ-regular and δ corresponds to the maximum gap size for the PMC algorithm, as described in Section 2.2.1. If representation Δ is transferred to another system without error, but the new system relies on a PMC algorithm that assumes a gap size $\delta' < \delta$, then it is straightforward to show that the virtual interval $[S'_-, S'_+]$ is still ϵ-regular in the new system as long as $S'_- \supseteq S_-$ and $S'_+ \subseteq S_+$ are maintained. In this case, no repair of the transferred model is needed, even though its topological form may differ from the original model. For other examples of how class-of-sets semantics may be used in the context of data translation, the reader is referred to [21].

In what follows, the differences between the two semantics will be demonstrated in the context of a certain PMC algorithm, which we describe in the following section.

3.2 Example: a PMC algorithm

It has been observed [22, 23] that PMC is a fundamental operation underlying many other operations in solid modeling. In this section we will prove statements about a PMC algorithm for extruded solids, a case which is general enough to be of practical interest, but simple enough to be transparent. We will use these results to illustrate the remarks of Section 3.1.

The algorithm discussed here is implemented using finite-precision floating-point arithmetic, is applicable to extruded solids (defined by means of generalized cylinders [24, Sec. 8.3]), and is a variant of an algorithm proposed by Nishita *et al* [25]. We will refer to this algorithm as the *NSK* algorithm; it is based on an idea (known to Poincaré [26, p. 49]) for computing the winding number, combined with linear separation and repeated subdivision.

The fact that our results apply only to parametric extruded solids means, of course, that we are really only dealing with a two-dimensional problem in disguise. Similar results, for more general solids, will require different formal machinery. On the other hand, the extruded-solid example is general enough to illustrate the trade-offs between the two semantical approaches.

In [24, Sec. 8.3], a generalized cylinder $\boldsymbol{S}$ is defined[4] as the Cartesian product of a simple closed planar NURBS curve $\boldsymbol{C}(u)$, $0 \leq u \leq 1$, and a closed interval on the real line. (We will suppose that the endpoints of the NURBS curve interpolate its initial and final control points.) Then, an *extruded solid* (based on the generalized cylinder), can be defined as

$$\{\boldsymbol{x} \in R^3 : (x_1, x_2) \in int(\boldsymbol{C}(u)), 0 < x_3 < d\}$$

where $int(\boldsymbol{C}(u))$ is the open interior of the bounded region, defined by $\boldsymbol{C}(u)$, whose existence is guaranteed by the Jordan curve theorem.

In this paper, however, rather than requiring that the cross-sections of the extruded solid be defined by a NURBS curve, we will permit instead a finite sequence of NURBS curves, $C_1(u), \ldots, C_n(u)$, forming a cycle, as illustrated in Figure 7. There may be gaps in the chain of curves: the dotted lines in Figure 7 are,

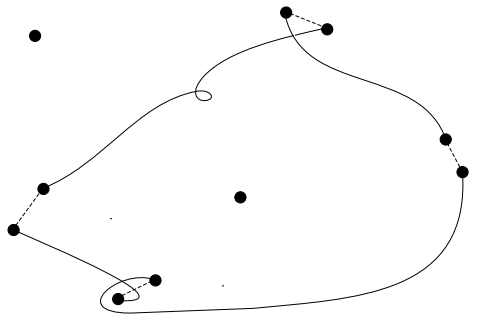

Figure 7: Cross-section of ruled-surface object

for now, intended only to logically link the ordered sequence of curves. There may also be self-intersections, and overlaps between segments $\boldsymbol{C}_i(u)$ and $\boldsymbol{C}_{i'}(u)$.

We begin by proving certain statements about the NSK algorithm, when applied to a *closed* (not necessarily simple) curve. Although this information is not needed immediately, we will mention now, for purposes of motivation, how we will later transform a set of curve segments like those of Figure 7 into a closed curve, so that the statements proved here can be applied. One possibility (which will be used for single-set semantics) is, for each of the dotted lines in Figure 7, to define $\boldsymbol{b}$ to be, say, the midpoint of the dotted line, and to use the Whitney Extension Theorem to adjust the adjacent curves so that they terminate at $\boldsymbol{b}$. A second possibility (which will be used for class-of-sets semantics) is to replace each dotted line by an actual straight-line segment. Both choices lead to a closed curve.

3.2.1 The Poincaré algorithm

Suppose first that each $\boldsymbol{C}_i(u)$ is a straight-line segment $(1-u)\boldsymbol{v}^{(i-1)} + u\boldsymbol{v}^{(i)}, 0 \leq u \leq 1, i = 1, \ldots, n$, where each $\boldsymbol{v}^{(i)}$ lies in the $x_1 - x_2$ plane. The *winding number* of $\boldsymbol{p}$ is defined as the sum over i of the signed angles between $\mathbf{v}^{(i)} - \mathbf{p}$ and $\mathbf{v}^{(i-1)} - \mathbf{p}$, divided by 2π. Poincaré [26] pointed out the now well-known fact that it is sufficient to count the number of times that the vector $\mathbf{v}^{(i)} - \mathbf{p}$ passes through a given predefined direction (say, $x_1 = 0, x_2 = 1$) in order to compute the winding number. (This version of the winding-number algorithm is closed related to the line-crossing algorithm [27, p. 965] for the point-in-polygon problem: the latter algorithm can be viewed as computing the winding number modulo 2.)

[4] Actually, in [24, Sec. 8.3], a slightly more general case is considered.

The angle through which $\mathbf{v} - \mathbf{p}$ turns as $\mathbf{v}$ moves along the polygon from $\mathbf{v}^{(i-1)}$ to $\mathbf{v}^{(i)}$ is defined to be the angle between the two vectors, measured positively in the counterclockwise direction, and chosen in the interval $[-\pi, \pi]$. Thus in Figure 8, if $\mathbf{u} = \mathbf{v}^{(i-1)}$ is the previous vertex, and $\mathbf{w} = \mathbf{v}^{(i)}$ is the current vertex,

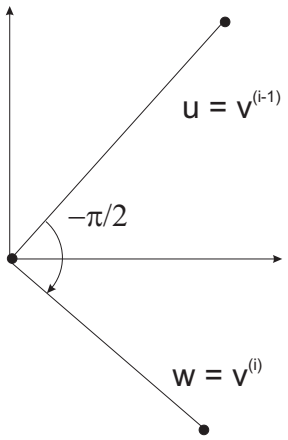

Figure 8: The angle from $\mathbf{u}$ to $\mathbf{w}$ is negative

then the angle between $\mathbf{u} - \mathbf{p}$ and $\mathbf{w} - \mathbf{p}$ is $-\pi/2$, and not $3\pi/2$. We omit discussion of the ambiguous case when the angle is $\pm\pi$, and the cases when $\boldsymbol{u} = \boldsymbol{p}$ or $\boldsymbol{w} = \boldsymbol{p}$, since these will not be needed below.

A *pole crossing* means that in following the boundary of the polygon from $\mathbf{v} = \mathbf{u} \equiv \mathbf{v}^{(i-1)}$ to $\mathbf{v} = \mathbf{w} \equiv \mathbf{v}^{(i)}$, $\mathbf{v} - \mathbf{p}$ passes through the positive x_2-axis (the "North Pole"). If one or the other (but not both) of $\mathbf{u}$ and $\mathbf{w}$ is on the positive x_2-axis, a half pole crossing is said to have occurred. In either case, if the angle from $\mathbf{u} - \mathbf{p}$ to $\mathbf{w} - \mathbf{p}$ is positive, we say that a *positive* pole crossing has occurred, and the algorithm increments the counter W by 1 (or by 1/2 if $w_1 - p_1 = 0$, $w_2 - p_2 > 0$ or if $u_1 - p_1 = 0$, $u_2 - p_2 > 0$); if the angle from $\mathbf{u} - \mathbf{p}$ to $\mathbf{w} - \mathbf{p}$ is negative, we say that a *negative* pole crossing has occurred, and the algorithm decrements the counter W by 1 (or by 1/2 if $w_1 - p_1 = 0$, $w_2 - p_2 > 0$ or if $u_1 - p_1 = 0$, $u_2 - p_2 > 0$).

Define $\mathbf{v}^{(n)} = \mathbf{v}^{(0)}$, and initialize the variable W to 0. For each $i = 1, ..., n$ let $\mathbf{u} \leftarrow \mathbf{v}^{(i-1)}$ and $\mathbf{w} \leftarrow \mathbf{v}^{(i)}$, and change the value of W (which at the end will contain the value of the winding number) according to the following.

If $u_1 - p_1 > 0$, there is a pole crossing if and only if

$$\begin{aligned} &\bullet\ w_1 - p_1 \leq 0 \quad \text{and} \\ &\bullet\ -(u_2 - p_2)(w_1 - p_1) + (u_1 - p_1)(w_2 - p_2) > 0. \end{aligned} \tag{5}$$

It is a half pole crossing if $w_1 = p_1$, in which case $W \leftarrow W + 1/2$; otherwise, $W \leftarrow W + 1$. This case is illustrated in Figure 9. Note that the case, where the second inequality displayed above is replaced by equality, cannot occur in our application, since we will not need to deal with the case where $\mathbf{p}$ is on the boundary.

If $u_1 - p_1 < 0$, there is a pole crossing if and only if

$$\begin{aligned} &\bullet\ w_1 - p_1 \geq 0 \quad \text{and} \\ &\bullet\ -(u_2 - p_2)(w_1 - p_1) + (u_1 - p_1)(w_2 - p_2) < 0. \end{aligned} \tag{6}$$

It is a half pole crossing if $w_1 = p_1$, in which case $W \leftarrow W - 1/2$; otherwise, $W \leftarrow W - 1$. This case is illustrated in Figure 10. Again, the case where the second inequality is replaced by equality cannot occur.

If $u_1 = p_1$, there is a half pole crossing if $w_1 - p_1 \neq 0$; in this case, $W \leftarrow W + 1/2$ if $w_1 - p_1 < 0$, and $W \leftarrow W - 1/2$ if $w_1 - p_1 > 0$. Note that is is *not* risky to check "$u_1 = p_1$?" using floating-point arithmetic, since the $\mathbf{v}^{(i)}$ and $\mathbf{p}$ are data items, uncontaminated by round-off error due to previous calculation. In particular, when $v_1^{(0)} - p_1 = 0$, the half pole crossing on initially leaving $\mathbf{v}^{(0)}$, and the half pole crossing on later arriving at $\mathbf{v}^{(n)} \equiv \mathbf{v}^{(0)}$ will both be recognized.

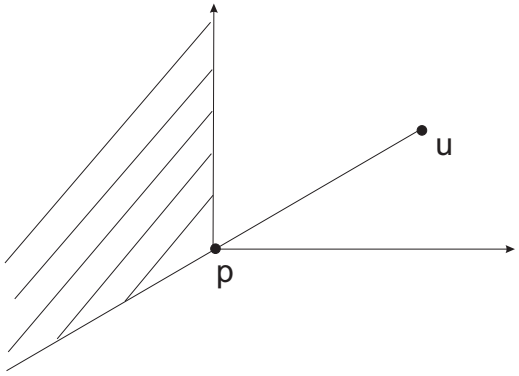

Figure 9: Zone for positive pole crossing

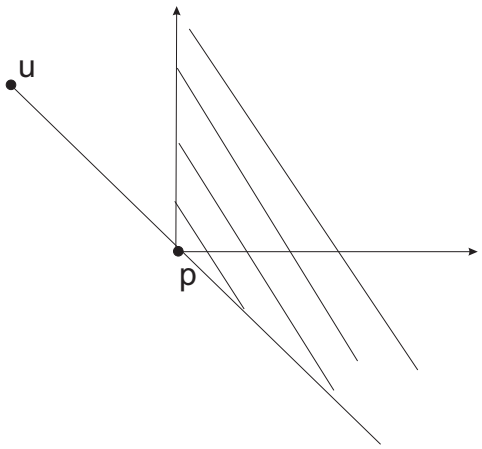

Figure 10: Zone for negative pole crossing

Neither is there any risk or difficulty in checking the sign of $w_1 - p_1$ in floating point, and therefore there is no worry about "almost" crossing the pole, but missing it, or "accidentally" crossing the pole when we should not.

There is, however, a potential difficulty in the evaluation of the inner products in (5) and (6). If the inner product is close to zero, the change in angle is in absolute value close to π, and it is possible to miss a pole crossing, or add a spurious one. Because of round-off error, the vector $\mathbf{v} - \mathbf{p}$ may be sent back in the wrong direction to an almost diametrically opposite point, and we would neglect to add (subtract) 1 to (from) W, or we would do so when we should not. This corresponds to a $\mathbf{p}$ which is very close to the boundary: see Figure 11.

This difficulty can be avoided in a fail-safe manner by means of the standard *a priori* bounds on floating-point arithmetic, *i.e.*, we can give sufficient conditions that, say, the sign of the inner product in (6) is correctly computed. Furthermore, the width of the region, near the line-segments in the polygon, whose status is ambiguous because the sufficient conditions are not applicable, is narrow. We note that the use of *a priori* bounds for other, similar purposes can be found in the computational-geometry literature [28, 29, 30].

We now give the derivation of the sufficient conditions. Let $fl[a \circ b]$ be the floating-point result of computing $a \circ b$, and let

$$M = \max\{|u_1 - p_1|, |u_2 - p_2|, |w_1 - p_1|, |w_2 - p_2|\}.$$

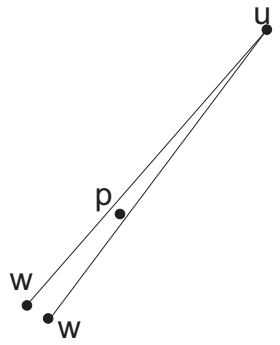

Figure 11: **p** close to the boundary

Then [19, Ch.3] we have[5], when verifying (6):

$$\begin{array}{l} fl[u_2 - p_2] = (u_2 - p_2)(1+\gamma_1) \\ fl[w_1 - p_1] = (w_1 - p_1)(1+\gamma_2) \\ fl[(u_2 - p_2)(w_1 - p_1)] = (u_2 - p_2)(w_1 - p_1)(1+\gamma_1)(1+\gamma_2)(1+\gamma_3) \\ fl[u_1 - p_1] = (u_1 - p_1)(1+\gamma_4) \\ fl[w_2 - p_2] = (w_2 - p_2)(1+\gamma_5) \\ fl[(u_1 - p_1)(w_2 - p_2)] = (u_1 - p_1)(w_2 - p_2)(1+\gamma_4)(1+\gamma_5)(1+\gamma_6) \end{array}$$

and

$$\begin{array}{l} fl[(u_1 - p_1)(w_2 - p_2) - (u_2 - p_2)(w_1 - p_1)] \\ = (u_1 - p_1)(w_2 - p_2)(1+\gamma_4)(1+\gamma_5)(1+\gamma_6)(1+\gamma_7) \\ -(u_2 - p_2)(w_1 - p_1)(1+\gamma_1)(1+\gamma_2)(1+\gamma_3)(1+\gamma_7) \quad . \end{array}$$

Here, $|\gamma_i| \leq \eta$, where η is the precision of the floating-point arithmetic being used (about 10^{-7} for IEEE single precision [20]). Consequently,

$$\begin{array}{l} |(u_1 - p_1)(w_2 - p_2) - (u_2 - p_2)(w_1 - p_1) \\ -fl[(u_1 - p_1)(w_2 - p_2) - (u_2 - p_2)(w_1 - p_1)]| \\ \leq 8M^2\eta + O(\eta^2). \end{array} \tag{7}$$

Furthermore, in practice [19, pp. 113-114], we can omit the $O(\eta^2)$ terms by making a small change in η, so that the bound (7) can be written as

$$8M^2\eta_o, \quad \eta_o \cong 10^{-7}.$$

Consequently, a sufficient condition that the second inequality in (6) is satisfied is that

$$fl[(u_1 - p_1)(w_2 - p_2) - (u_2 - p_2)(w_1 - p_1)] < -8M^2\eta_o,$$

and a sufficient condition that it is *not* satisfied is that

$$fl[(u_1 - p_1)(w_2 - p_2) - (u_2 - p_2)(w_1 - p_1)] > 8M^2\eta_o.$$

Suppose, on the other hand, that $u_1 - p_1 < 0$ and $w_1 - p_1 \geq 0$, but the second inequality in (6) cannot be verified or excluded, since $|[fl[(u_1 - p_1)(w_2 - p_2) - (u_2 - p_2)(w_1 - p_1)]| \leq 8M^2\eta_o$. One or the other of the two vectors

$$\begin{bmatrix} -(u_2 - p_2) \\ u_1 - p_1 \end{bmatrix}, \qquad \begin{bmatrix} -(w_2 - p_2) \\ w_1 - p_1 \end{bmatrix}$$

[5]The notation fl refers to the actual value computed using floating-point arithmetic. There is an abuse of notation in the displayed quantities: it is assumed that the method of calculation of combined quantities is understood. Thus, for example, $fl[(u_2 - p_2)(w_1 - p_1)]$ is shorthand for $fl[fl[u_2 - p_2] \cdot [fl[w_1 - p_1]]$.

has norm at least M. Without loss of generality, suppose that it is the first of these vectors, and let

$$\boldsymbol{n} = \begin{bmatrix} -(u_2 - p_2) \\ u_1 - p_1 \end{bmatrix}.$$

The maximum distance from $\boldsymbol{p}$ to the line-segment $\boldsymbol{u} - \boldsymbol{w}$ is $\sqrt{2}\left|\frac{\boldsymbol{n}}{\|\boldsymbol{n}\|} \cdot (\boldsymbol{w} - \boldsymbol{p})\right|$. We have $\|\boldsymbol{n}\| \geq M$, and

$$\sqrt{2}\left|\frac{\boldsymbol{n}}{\|\boldsymbol{n}\|} \cdot (\boldsymbol{w} - \boldsymbol{p})\right| \leq \frac{\sqrt{2}}{M}|-(u_2 - p_2)(w_1 - p_1) + (u_1 - p_1)(w_2 - p_2)| \leq 16\sqrt{2}M\eta_o.$$

A similar analysis applies to the computation of the second inequality in (5).

3.2.2 The NSK algorithm

Let us now consider the case of curve segments that are not straight lines, but, instead, more general curves such as those illustrated in Figure 7. The fundamental idea of the NSK algorithm is that this more general case can be reduced to the straight-line case, in the following way. Suppose that the convex hull of, say, the top NURBS segment (the one containing a loop) in Figure 7 can be separated from a point $\boldsymbol{p}$ by a straight line. Then, for the purposes of computing the winding number, a NURBS segment whose endpoints interpolate its initial and final control points can be replaced by a straight line joining the endpoints: whatever the number of pole crossings for the NURBS segment, the net total will be -1, 0 or 1, the number of crossings for the straight-line segment joining the endpoints of the NURBS segment.

If it is not possible to separate a certain NURBS segment from $\boldsymbol{p}$, then the NSK algorithm recursively subdivides the segment, placing one part of the segment on a stack, and recursively attempting to separate the other part of the segment from $\boldsymbol{p}$. Suppose that the stopping criterion for this recursive subdivision process is such that the distance from $\boldsymbol{p}$ to the original NURBS segment is less than or equal to σ. (Note that the derivation of the constant σ will itself involve use of the *a priori* bounds on floating-point operations. We omit this derivation.) Then we can guarantee that the NSK algorithm will produce the correct value of the winding number for all points $\boldsymbol{p}$ at distance at least

$$\max\{\sigma, 22.8M\eta_o\}$$

from the NURBS segments forming the cross-section of the ruled solid, and that it will either produce the correct answer or the output "don't know" for points closer than this. This is done by returning "don't know" if (5) or (6) ever fails to produce a guaranteed answer, or if the subdivision algorithm stops because it is unable to separate $\boldsymbol{p}$ from a boundary segment. The algorithm never produces an incorrect answer, and points of uncertainty are all within $\max\{\sigma, 22.8M\eta_o\}$ of the boundary curve.

3.2.3 The algorithm in the context of single-set semantics

Consider the following class of extruded solids. The boundary patches (see Section 2.1) of the solid are defined by

$$\boldsymbol{F}_i(u, v) = \{(\boldsymbol{C}_i(u), v) : 0 \leq u, v \leq 1\},$$

where the $\boldsymbol{C}_i(u)$ are NURBS curves, $i = 1, \ldots, n$, as described at the beginning of Section 3.2. Thus, the domain D of each patch is $D_0 = [0, 1]^2$, and the patches are not trimmed. The explicit boundary curves are defined by $(\boldsymbol{C}_i(u), 0)$ and $(\boldsymbol{C}_i(u), 1)$, $0 \leq u \leq 1$, along each side of the patch, and by a constant function (independent of u) along the ends of the patch. (For example, we might have

$$\boldsymbol{b}_i(v) = \left(\frac{1}{2}[\boldsymbol{C}_{i-1}(1) + \boldsymbol{C}_i(0)], v\right), 0 \leq v \leq 1,$$

for $i = 1, \ldots, n$, with $C_0(u) \equiv C_n(u)$.) We suppose that

$$\max\{\|\boldsymbol{b}_i(v) - (\boldsymbol{C}_{i-1}(1), v)\|, \|\boldsymbol{b}_i(v) - (\boldsymbol{C}_i(0), v)\|\} \leq \tau,$$

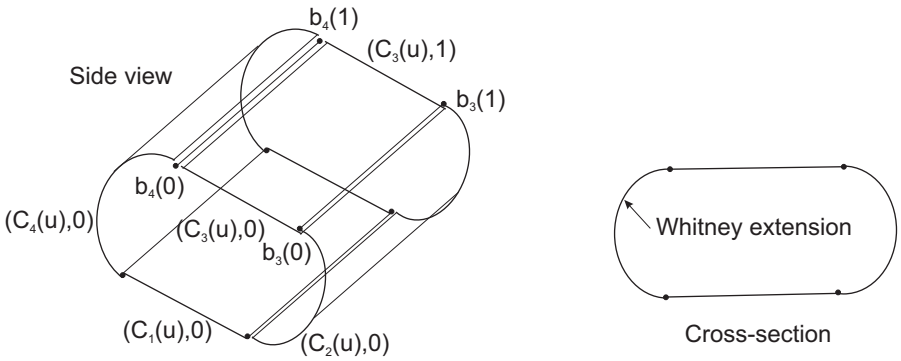

Figure 12: Example extruded solid

where $\tau > 0$, and that the pair of explicit vertices (see Section 2.1) at the corners of a patch are equal to $\boldsymbol{b}_i(0)$ and $\boldsymbol{b}_i(1)$, respectively. See Figure 12.

It is important to notice, however, that many requirements must be fulfilled [7] in order for a collection of data Δ to be well-formed, and define a subset S of R^3 such that $S \models \Delta$. In particular, the data illustrated in Figure 7 would not qualify, because of the self-intersection of one of the patches, and because of extraneous intersections between adjacent patches, which are supposed to intersect only along their boundaries. (Verification of these conditions is discussed in [7].)

Suppose, on the other hand, that the data satisfies the conditions necessary to guarantee that $S \models \Delta$. With the single-set semantics of [7], the cross-sections $\boldsymbol{C}_{i-1}$ and $\boldsymbol{C}_i$ of two neighboring faces will be adjusted, by using the Whitney Extension Theorem, so that they coincide along the explicit constant boundary curve $\boldsymbol{b}_i(v)$ (see cross-section in Figure 12). If the adjusted face is $\boldsymbol{F}_i(u,v)+\boldsymbol{\epsilon}_i(u,v)$, it can be shown that $\|\boldsymbol{\epsilon}_i(u,v)\|$ is smaller than τ (and also that $\boldsymbol{\epsilon}_i$ satisfies a Lipschitz condition). We would therefore like to modify the tests (5) and (6) in a way that guarantees that the algorithm will return "don't know" if the distance from $\boldsymbol{p}$ to the line-segment $\boldsymbol{u}-\boldsymbol{w}$ is less than or equal to τ. This can be done in the subdivision step by requiring a separation of at least τ between the convex hull of the NURBS segment and the point $\boldsymbol{p}$. (As a consequence, the stopping criterion σ must be at least as large as τ.) Thus, if we have the structure necessary to guarantee that $S \models \Delta$, we can add a fail-safe PMC algorithm to the list of methods (see the first item in the list at the beginning of Section 3.1) that can be rigorously applied to Δ.

3.2.4 The algorithm in the context of class-of-sets semantics

Suppose now that we do not have the structure of Section 2.1, involving approximate trimmed NURBS patches. As mentioned in Section 2.2.3, it may happen, in practice, that we have very little information at all about the structure of the boundary elements. In this section, however, we consider a case where a fair amount of structure is available, although not enough to guarantee that $S \models \Delta$. Thus, this is a simple application of class-of-sets semantics, in a very special case. But it is sufficiently general to permit comparison between the single-set and class-of-sets approaches.

Suppose that the data is a collection of faces, as described in Section 3.2.3, but with unspecified gaps, loops and overlaps, as illustrated in Figure 7. We suppose also that no reliable information is available about explicit boundary curves $\boldsymbol{b}$, or explicit vertices $\boldsymbol{v}$. It may nonetheless be true that the user is quite sure that the data defines an extruded solid, and that the level of precision is at least, say, δ_o. Finally, the user may believe, in practice, that the NSK algorithm applied to such data will provide a reliable result, so that it is reasonable to assume that the given data defines a ϵ-regular virtual interval by means of (3). The class-of-sets semantics provides a way to formulate a theorem supporting this intuition.

In order to proceed, we must introduce a hypothesis which ensures that the user has at least provided data that can be interpreted as a simple closed curve. As already mentioned, for practical applications we want to deal with situations like the one illustrated in Figure 7, where imperfections in the boundary are permitted. But if we put no limits at all on the size of gaps, loops and overlaps, then the user could provide

data impossible to interpret as the boundary of a generalized cylinder. Where, for example, is the interior of each of the three sequences of line segments shown in Figure 13? (In the first two sequences, which define

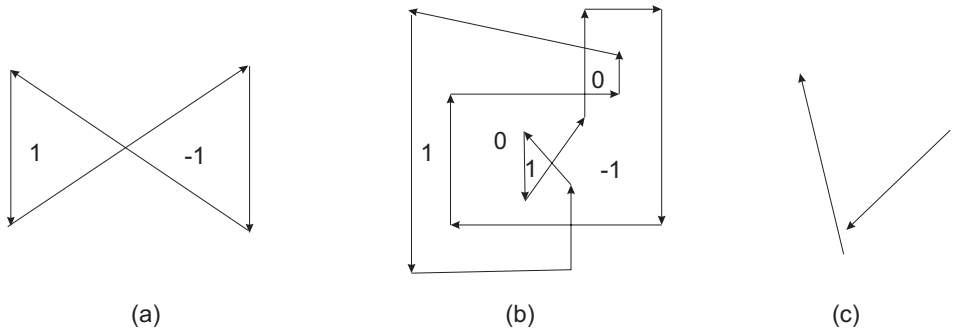

Figure 13: Three nonsense sequences

closed paths in R^2, the winding numbers are also given in Figure 13.) We introduce the hypothesis imposing the mentioned limits immediately, in order to emphasize that it is there not for the purpose of dealing with the effects of finite-precision arithmetic, but, rather, because we need data that can be interpreted as similar to a simple closed curve (but possibly containing imperfections).

The hypothesis is this. Given the ordered set of segments provided by the user, convert it to a closed path by joining the endpoint of each edge to the starting point of its successor, by a straight-line segment, unless the endpoint and starting point are already coincident. This is done in cyclic fashion for all given segments, and the resulting closed path is assumed to have n segments, including any new ones added by the process just described. Let the ordered set of vertices in this path be denoted $\boldsymbol{v}^1, \ldots \boldsymbol{v}^n$, and let $\boldsymbol{C}_i$ be the curve between $\boldsymbol{v}^{(i)}$ and $\boldsymbol{v}^{(i+1)}$, $i = 1, \ldots, n$, with indices computed modulo n. We can now apply the NSK algorithm as in Section 3.2.3 to induce the virtual interval $[S_-, S_+]$ that is ϵ-regular by assumption.

This, however, is not enough to guarantee that the algorithm will produce results consistent with users' expectations. The above construction is justified based on the expectation that the loops, overlaps and added line segments will be contained within the ϵ-regular interval, but we have no way to guarantee that NSK algorithm will produce the same results when applied to the boundaries of S_-, S_+, and the corrected data. Thus, in addition, we assume that the user has provided an upper bound $\delta_o > 0$ for the imperfections in the boundary data, in the following sense.

We suppose first that the boundary of the set

$$\cup_{i=1,\ldots,n} \cup_{\boldsymbol{x} \in \boldsymbol{C}_i} B_{\delta_o}(\boldsymbol{x})$$

is composed of two nested simple closed curves in the plane. The resulting nested simple closed curves define (by two applications of the Jordan curve theorem) a bounded set and an unbounded set which, together with an intermediate annular region, cover the plane (see Figure 14). We also suppose that the winding number for points in the interior bounded region has absolute value equal to 1, and the winding number for points in the unbounded region is equal to 0.

It is clear that with this hypothesis, the NSK algorithm works correctly, as in Section 3.2.3, if we replace the constant τ by $\delta_o/2$. Indeed, the algorithm produces an interval $[S_-, S_+]$ satisfying (2), with $S_- = S_{in}$, $S_+ = S_{out}$. The interval is assumed to be ϵ-regular, and the points on boundaries of S_- and S_+ lie at most $\max\{\sigma(\delta_o), 22.8M\eta_o\}$ away from the original NURBS segments in the boundary of the curve's cross-section, with $\delta_o \leq \sigma(\delta_o)$. Thus, the intermediate annular region acts as a sort of black box, in which we can hide all of the inconsistencies illustrated in Figure 7, and all of the low-level vagaries of finite-precision arithmetic (as illustrated, for example, in [31, Figure 1]).

It is important, of course, that the user be aware of the implications of the hypotheses. For example, consider the case of a split-ring for the piston of an automobile engine. The hypothesis concerning the two nested simple closed curves, illustrated in Figure 14, may be satisfied for the split-ring, in its rest state outside the engine cylinder, for a certain value of δ_o, as illustrated in Figure 15. In this case, the implied

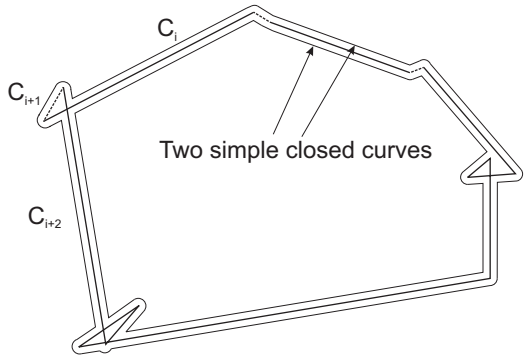

Figure 14: Two nested simple closed curves

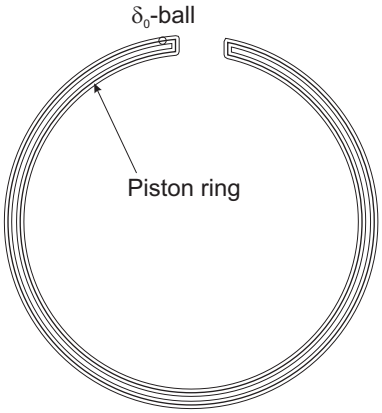

Figure 15: Nested curves for split-ring

interval $[S_-, S_+]$ is also δ_o-regular. On the other hand, if δ_o is increased slightly, the interval could remain δ_o-regular, while at the same time the hypothesis that the winding number is correctly computed using the NURBS curves $\boldsymbol{C}_i$ fails. Similarly, if the split-ring is moved to its compressed state within the engine cylinder, then both hypotheses about the winding number and δ_o-regularity fail. It remains true, however, that if we do not have the hypotheses necessary to support the use of single-set semantics, but we can justify the hypotheses of this section, then we can still prove a useful result, using the class-of-sets semantics.

4 Conclusion

The main goal and contribution of this paper is a thorough examination of the semantic foundations of the so-called robustness problem in geometric modeling. The problem is very different from robustness problems arising in other fields, such as computational geometry. In geometric modeling, the main issue is not the magnitude and propagation of approximation and computation errors. The mere presence of *any* errors in geometric data and/or algorithms undermines the *assumed validity* of all subsequent computations and applications using the data. Since virtually all models and algorithms contain errors and approximations, development and standardization of alternative semantics of such models is critical to both theoretical and practical development of geometric modeling as a discipline.

In this paper, we analyzed and compared two possible approaches for assigning a meaning to geometric models in the presence of such errors: the single-set semantics interprets the data as a slight adjustment of the original data that preserves the topological form, and the class-of-sets semantics interprets the data as a family of sets that are close to each other but may possess distinct topological structure. The two

approaches have been shown to be complementary. The single-set semantics assumes and enforces a stable topological and combinatorial structure, and provides as well a strong measure of geometric similarity, under small perturbations. This semantics requires *a priori* satisfaction of strong conditions (compared to the class-of-sets approach), but it also provides the ability to prove the strongest possible statements about the use of the given geometric data. The single-set semantics is likely to be the semantics of choice within a single system environment. In contrast, the class-of-sets semantics puts very few restrictions on the geometric data, tolerates a much wider class of errors and uncertainties, and supports multiple interpretations of the same data. As such, the class-of-sets semantics may be most useful in situations involving multiple systems and/or uses of data, such as those arising in data translation and collaborative environments. While the class-of-sets semantics implies strong metric properties, it neither preserves nor guarantees the topological form of the represented set, and requires additional hypotheses or assumptions about the source or the use of data.

We have demonstrated the two approaches and their complementary properties in the context of a particular popular PMC algorithm that underlies many other geometric computations. In the process, we made a number of specific decisions (for example, on geometric predicates) and assumptions (for example, on closing the gaps between the adjacent vertices). We do not claim that these choices are optimal, or even preferable to other possibilities. Their sole purpose is to explain the similarities and differences in the two possible methods of assigning semantics to geometric models, which we expect will persist and extend to other types of applications and algorithms. It should be clear that two approaches are *not* mutually exclusive and may be used simultaneously, provided the conditions required by each approach are satisfied. In cases where it is appropriate to make both the assumptions of Section 2.1, *and* assumptions like those of Section 2.2, then similar results will be provable in the single-set and class-of-sets semantics. That is, the result in each semantics will have essentially the same content, and the proofs will be similar.

More generally, it may be observed that a number of choices can be made in the definition of $\models$. For example, the single-set semantics requires that the set S should have exactly the combinatorial structure given in the topological data. The class-of-sets semantics gives precedence to the geometric data, rather than the topological data.[6] The correct choice depends on which theorems are to be proved, and this may depend on the downstream applications that are to be considered. In general, strengthening the conditions on the set S will tend to increase the need for supplementary hypotheses in theorems, and increase the difficulty of proving the theorems; this will, however, be justified to the extent that it provides a more useful model of the real world. Similarly, weakening the conditions on S will tend to reduce the requirements for supplementary hypotheses, and reduce the difficulty of proving theorems, and this will be justified to the extent that it simplifies and clarifies.

5 Acknowledgements

The use of half pole crossings (Section 3.2.1), which has several technical advantages over including the positive x_2-axis within one of its neighboring quadrants, was suggested by J.-F. Rotgé.

The work of the first two authors was supported in part by the National Science Foundation grants DMI-0500380, DMI-0323514, and CCR-0112758. The research of the third author was supported in part by a grant from the Natural Sciences and Engineering Research Council of Canada.

References

[1] Requicha, A. A. G. Representations for rigid solids: theory, methods and systems. *Computing Surveys* 1980; 12(4):437-464.

[2] Requicha, A. A. G. Toward a theory of geometric tolerancing. *Int. J. Robot. Res.* 1983; 2(4):45-60.

[3] Boyer, M. and Stewart, N. F. Imperfect form tolerancing on manifold objects: a metric approach. *Int. J. Robot. Res.* 1992; 11(5):482-90.

[6] This question has been discussed often in the literature on robustness in solid modeling [32, 33], but in [7, Sec. 3.2], strong arguments in favor of fixing the topological structure were given.

[4] Lear, D. A. Practical applications of algebraic topology to geometric design. Working paper, IFIP WG5.2, Workshop notes, Tab. 3. Fourth IFIP WG5.2 Workshop on Geometric Modeling and Computer-Aided Design, Rensselaerville, NY, Sept. 27 - Oct. 1, 1992.

[5] Rossignac, J. R. and O'Connor, M. A. SGC: A dimension-independent model for pointsets with internal structures and incomplete boundaries. In: Geometric Modeling for Product Engineering, IFIP WG 5.2/NSF Working Conference on Geometric Modeling, Rensselaerville, NY, North Holland, 1990, p. 145-80.

[6] A. Biswas, V. Shapiro, and I. Tsukanov. Heterogeneous material modeling with distance fields. *Computer Aided Geometric Design*, 2004; 21(3):215-242.

[7] Andersson, L.-E., Stewart, N. F. and Zidani, M. Error analysis for operations in solid modeling. Unpublished results (manuscript, 34 pages). Available at *www.iro.umontreal.ca/~stewart*.

[8] Hoffmann, C. M. and Stewart, N. F. Accuracy and semantics in shape-interrogation applications. In press, *Graphical Models* 2005.

[9] Qi, J. and Shapiro, V. Epsilon-regular sets and intervals. Proceedings, Int'l. Conf. on Shape Modeling and Applications, MIT, Cambridge, MA, 15-17 June, 2005, 308-17.

[10] Litke, N., Levin, A. and Schröder, P. Trimming for subdivision surfaces. *Comp. Aided Geom. Design* 2001; 18:463-81.

[11] Saia, A., Bloor, M. S., de Pennington, A. Sculptured surface shapes using inner and outer bounded models. Proceedings IMA88, 1988, 451-72.

[12] Sakkalis, T., Shen, G. and Patrikalakis, N. M. Topological and geometric properties of interval solid models. *Graphical Models* 2001; 63:163-75.

[13] ACIS 3D Toolkit, Spatial Technology, Boulder, CO, 1998.

[14] STEP International Standard, ISO 10303-42, Industrial automation systems and integration — Product data representation and exchange — Part 42: Integrated generic resources: Geometric and topological representation, International Organization for Standardization (ISO), Reference Number ISO 10303-42: 1994(E), First Edition 1994-12-15, 1997.

[15] Whitney, H. Analytic extensions of differentiable functions defined on closed sets. *Trans. Am. Math. Soc.* 1934; 36:63-89.

[16] Stam, J. On subdivision schemes generalizing uniform B-spline surfaces of arbitrary degree. *Comp. Aided Geom. Design* 2001; 18(5):383-96.

[17] Stewart, I. F. and Foisy, A. R. Arbitrary-degree subdivision with creases and attributes. *J. of Graphics Tools* 2004; 9(4):3-17.

[18] Jayaraman, R. and Srinivasan, V. Geometric tolerancing I. Virtual boundary requirements. *IBM J. of Research and Development* 1989; 33(2):90-104.

[19] Wilkinson, J. H. The Algebraic Eigenvalue Problem. Oxford University Press: Oxford, 1965.

[20] IEEE Standard for Binary Floating-Point Arithmetic. ANSI/IEEE Std 754-1985, IEEE, New York, NY, 1985.

[21] Qi, J. and Shapiro,V. Epsilon-Solidity in Geometric Data Translation. Technical Report SAL 2002-4, Spatial Automation Laboratory, University of Wisconsin-Madison, June 2004.

[22] Shapiro, V. Solid modeling. In G. Farin, J. Hoschek, and M.-S. Kim, editors, *Handbook of Computer Aided Geometric Design*. Elsevier Science Publishers, 2002.

[23] Tilove, R. B. Set membership classification: a unified approach to geometric intersection problems. *IEEE Transactions on Computers* 1980; C-29(10):874-83.

[24] Piegl, L. and Tiller, W. The NURBS Book. Berlin: Springer, 1997.

[25] Nishita, T., Sederberg, T. and Kakimoto, M. Ray tracing trimmed rational surface patches. *Computer Graphics* 1990; 24(4):337-45.

[26] Henle, M. A Combinatorial Introduction to Topology. San Francisco: Freeman, 1979.

[27] Foley, J. D., van Dam, A., Feiner, S. K. and Hughes, J. F. Computer Graphics (Second Edition). Reading: Addison-Wesley, 1990.

[28] L. Guibas, D. Salesin and J. Stolfi. Epsilon geometry: Building robust algorithms from imprecise computations. In Proc. of the Fifth ACM Symposium on Computational Geometry, Saarbruchen, West Germany, 1989, 208–17.

[29] L. Kettner and E. Welzl. One sided error predicates in geometric computing. In Proc. 15th IFIP World Computer Congress, Fundamentals - Foundations of Computer Science, Aug. 1998, 13–26.

[30] J. R. Shewchuk. Adaptive precision Floating-point arithmetic and fast robust geometric predicates. *Discrete & Computational Geometry* 1997; 18(3):305–363.

[31] Kettner, L., Mehlhorn, K., Pion, S., Schirra, S. and Yap, C. Classroom examples of robustness problems in geometric computations. In Proc. of the 12th Annu. European Sympos. Algorithms (ESA'04), LNCS 3221, Springer-Verlag, Bergen, Norway, Sep. 2004, 702–13.

[32] Hopcroft, J. E. and Kahn, P. J. A paradigm for robust geometric algorithms. Technical Report TR89-1044, Department of Computer Science, Cornell University, Ithaca, NY, 1989.

[33] Hoffmann, C. M., Hopcroft, J. E. and Karasick, M. S. Towards implementing robust geometric computations. Proc. of the Fourth Annual ACM Conference on Computational Geometry, Urbana-Champaign, IL, 1988, 106-17.