ε-Solidity in Geometric Data Translation

Jianchang Qi   Vadim Shapiro
Spatial Automation Laboratory *
University of Wisconsin-Madison

August 3, 2004

Abstract

Classical theory of solid modeling relies on the notion of regular sets and presupposes exactness in both geometric data and algorithms. In contrast, modeling, exchange and translation of geometric models in engineering applications usually involve data approximations and algorithms with different numerical precisions. We argue that an appropriate formulation of the geometric data translation problems requires finite size neighborhoods, leading to the notion of ε-topological operations. These operations are then used to formulate the definitions of ε-regularity and ε-solidity that extend and subsume the corresponding classical concepts as exact special cases. The proposed theory allows systematic classification and investigation of problems in geometric data translation. In particular, it explains why the current methods for validity checking of boundary representations are neither necessary nor sufficient for maintaining ε-solidity in the presence of numerical inaccuracies, whereas geometric healing procedures may be avoided in many common situations. Furthermore, the proposed theory suggests how the classical solid modeling paradigm should be extended in order to deal with the outstanding problems in geometric robustness, validation, and data translation.

1 Introduction

1.1 Motivation

Computer aided design (CAD) data translation, especially solid model translation, has been a challenging problem for both industry and academia. Ability to exchange and translate data with no or little remodeling effort is a critical component of any scenario where design, manufacturing, and analysis applications share geometric models in a truly collaborative fashion. Despite the recent progress, geometric data interoperability between different systems remains an elusive goal, costing industry substantial amounts of time and money [15]. A typical geometric data translation problem between two systems is illustrated in Figure 1. A geometric representation can be thought of as a composition of geometric primitives by rules specific to a given representation scheme. In data translation, such a representation is transferred explicitly by various translators. However, the meaning of any representation is determined by the corresponding evaluation algorithms that usually also differ from system to system. Therefore, it is reasonable to assume that the evaluation algorithms are also transferred implicitly.[1] The scenario in Figure 1 subsumes many other types of translation problems. For example, the classical problems of boundary evaluation, boundary to CSG conversion, and other types of representation conversions correspond to the cases when the translations apply also to representation rules but usually take place within one common system.

Perhaps the most widespread difficulty arises from mismatch between the precision of geometric representation and the precision of the evaluation algorithms used in a modeling system. For example, if the sending and receiving systems rely on different precisions, the points on surface intersections may classify differently (ON or OFF) in the two systems. As the result of such a data translation, many design, manufacturing, and analysis tasks cannot be performed in the receiving system until the geometric models are either corrected ("healed") or remodeled. It is widely believed that many of the geometric data translation difficulties can be alleviated or bypassed altogether if the geometric representations, and rules in particular, are sufficiently high-level. Translation of parametric feature-based representations is particularly promising[51], because such representations are largely symbolic structures with

*Complete address: 1513 University Avenue, Madison, WI 53706, USA. E-mails: jian@sal-cnc.me.wisc.edu, vshapiro@engr.wisc.edu

[1]See Section 2.2 for additional discussion and examples.

minimal numerical data. However, success of this approach hinges on existence of standard and formal semantics of parametric and feature-based representations, including rules for determining boundaries of represented models and valid ranges of parameters. Development of such semantics is an active area of research [46, 27, 18, 48, 39, 6, 38, 10, 9, 52], but as of this writing, acceptable formal models are lacking in a number of important areas, including blending, persistent referencing, constraints, and validity, to name a few.

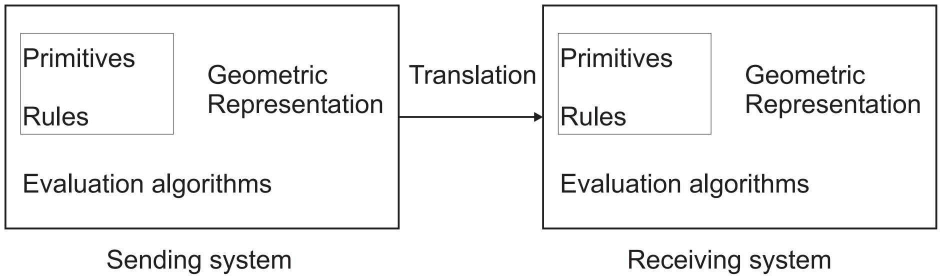


Figure 1: A generic geometric data translation diagram

Engineering applications require that all data translations result in solutions that are valid and consistent with intended use by the receiving system. Therefore all such solutions must be based on sound formal principles. Most currently proposed methods for dealing with data translation rely on the theoretical foundations laid out by Requicha over twenty years ago[40]. Specifically, it is widely accepted that a suitable model for a solid object is an r-set, defined as bounded, semi-analytic, and closed regular subset of E^3. The intuitive notion that every non-trivial solid has a non-empty interior and a thin boundary is formally captured by requiring that $X = ki(X)$, where k, i denote respectively the topological closure and interior of a set X. With this terminology, a geometric representation is deemed *valid* if it corresponds to at least one r-set, and two representations are *consistent* if they represent the same set of points. The purpose of exact representation conversions is to produce representations that are valid and consistent in accordance with this theory [40].

Unfortunately, the above definitions do not apply to modeling and translation problems in the presence of numerical errors or approximations. The classical solid modeling theory assumes that all sets of points and functions may be represented exactly by data structures and algorithms. We know that this is not true, but the issues of errors and precision are delegated to "robust" geometric computations or practical implementation issues for system designers. It should not be surprising that most of the proposed translation solutions are either very limited or provide *ad hoc* heuristic solutions without any guarantees. A common business practice to alleviate the translation problems altogether is to standardize on a single geometric modeling kernel, but this practice appears to be expensive, limiting, and unacceptable for many applications. Use of exact computations is not helpful because exact computations make no sense with much of the engineering data that is inherently imprecise. Lacking fundamental remedies to the translation problem, the industry has embraced "healing" (CAD data repair) as a method for ensuring the quality of the model in a receiving system. But such modifications of the original geometry are dangerous, expensive, and provide no guarantees. We discuss these approaches and other related research in Section 1.3.

The fundamental difficulty with all proposed solutions is that they are not based on a suitable formal model. In particular, we will demonstrate in Section 2 that the classical notions of solid representation validity and consistency (based on assumption of exactness) are too simplistic for data translation problem. As implied by the diagram in Figure 1, a proper formal model for data translation must explicitly account for three sources of errors: approximations of geometric primitives, different precisions of evaluation algorithms, and inconsistent or ill-defined rules. In this paper, we propose a theory focusing on the approximation and precision errors and apply it to several specific instances of the data translation problem.

1.2 Approach and Outline

After briefly summarizing in Section 1.3 previous efforts on geometric data translation, in Section 2 we formulate the general data translation problem based on several real world examples. We define validity and consistency in the context of data translation, and focus on the issues of validity in the rest of this paper. Specifically, in this paper, we are interested in the most common case of data translation where the representation rules are assumed to be the same in the sending and the receiving systems. Thus, we assume that boundary representations are translated into boundary representations, CSG representations to CSG representations, and so on. This means that translation problems may arise only from approximation of primitives and/or changes in the precision of evaluation algorithms.

In order to deal with inexactness of data and algorithms, in Section 3, we generalize the classical set topological notions of *interior* i, *closure* k, and *boundary* ∂ in terms of finite size neighborhoods. The size of a neighborhood ε relates to the precision of data and algorithms; when precision is limited, the size must be finite. The resulting generalized epsilon-topological operations i_ε, k_ε, and ∂_ε satisfy many of the classical topological properties, and provide the basis for formal definitions of ε-regular sets and ε-solids that explicitly recognize the role of imprecision. The classical r-set solid model proposed by Requicha [40] is shown to be a special case where $\varepsilon = 0$ corresponds to arbitrarily small neighborhoods.

Practical consequences of the proposed theory should become clear in Section 4. We show that, under the proposed model, popular solid validity-checking procedures are neither necessary nor sufficient when either solid representation or evaluation algorithms are limited in precision. We then examine common translation problems and discuss how validity may be maintained in each case without healing.

Open issues and promising extensions of the proposed approach are considered in section 5. We also discuss the broader implications of the proposed formal model of ε-solidity and its relation to classical solid modeling, geometric robustness, and other theories.

1.3 Related Work

A vast amount of literature exists on subject of geometric robustness that also deals with issues of data errors and algorithm precision. In this sense, CAD data translation appears to be a special of the general geometric robustness problem. But there are also important differences. For example, exact computation is one popular technique originally proposed to address the robustness problem in geometric computations [56, 23, 29]. The basic philosophy of this approach is to generate exact output model from exact input model by using exact integer arithmetic, rational arithmetic, or algebraic computations. But this philosophy is not practical in the context of CAD data translation, because most engineering models are intrinsically imprecise, and many useful engineering computations cannot be represented using exact arithmetic. Furthermore, CAD data translation is a relative simple one-step[2] problem, while the general robustness problems must assume that an output of geometric computations is used as the input for geometric operations iteratively. In addition, CAD data translation usually involves two modeling systems with different precisions while robust computations only involve a single 'native' modeling system.

Several other proposals attempted to deal with imprecision of data and/or algorithms. Notably, Guibas, Salesin, and Stolfi [25] proposed an Epsilon Geometry framework for building robust geometric algorithms out of imprecise computations that arise from the use of finite precision arithmetic. Their results are not applicable to the solid validity and translation problems because they assume that geometric data is exact and because they only deal with computational geometry predicates (such as coincidence, collinearity, point inclusion in convex polygon, and convexity). In [20, 21], Edalat and Lieutier proposed the domain solid model to extend the classical r-set model in solid modeling.[3] The underlying philosophy of their model is to use two disjoint open subsets (A, B) called *partial solid* to capture the interior and exterior of a classical solid X at finite stage of computation by using dyadic voxels or rational polyhedron. The formulation does not appear to provide any mechanism to connect the theory to practical representations and algorithms in solid modeling. A related notion of an *approximate interval solid*[44] arises naturally from numerical considerations and was used to repair defective geometric models[49] and to develop robust algorithms for intersection of surfaces[36].

Below, we briefly review previous work that is specifically related to the translation problem. Broadly, these efforts can be categorized as *tolerant computing* and geometric *healing.* Use of numerical tolerances in solid modeling has been advocated by many in order to improve the robustness of modeling computations [45, 22, 28], and consequently, to allow use of imprecise data and different precision in the receiving system. Typically, numerical tolerances are assigned to vertices, edges and faces in a solid boundary representation, and inferred geometric tolerance zones are used to maintain the relationships between the geometric entities. After every computation, the tolerances are updated incrementally in an attempt to maintain the consistency between the geometric data and the topological (combinato-

[2] In other words, every translation can be viewed as a unit process that may or may not introduce additional errors.
[3] As there are inconsistencies between the two papers, we refer the newer paper [21] for discussion.

rial) part of a boundary representation. Implied by all tolerant modeling approaches is the notion that a toleranced representation is valid if **there exists an exact r-set** whose boundary has the same combinatorial structure and lies within the tolerance zones. However, such a definition of validity is not practically verifiable, and the proposed approaches differ in the proposed heuristic algorithms for deciding on validity. For example, it is shown in [45, 22] that some choices of tolerance values may lead the proposed algorithm to the erroneous answers. Commercial software systems such as ACIS [3] and Parasolid [35] rely on tolerant modeling and suggest default tolerances for improved reliability, but do not offer any deterministic rules for maintaining tolerances or provide guarantees of validity.

In the absence of guaranteed solutions with toleranced models, both industry and academia embraced 'geometric healing' or CAD data repair as the only pragmatic solution to the data translation problem. The idea appears straightforward: since we know that the original data was valid in the sending system, and that translation may have introduced some small changes in data or algorithm precision, it should be possible to fix the model in the receiving system by making small changes in the representation. In essence, geometric repair is another attempt to find that one r-set model that guarantees solidity. A variety of repair procedures have been proposed for linear polyhedral models [8, 7, 13, 12, 34] as well as for more general solids [50, 49, 54]. Perturbation techniques [7, 50, 49] attempt the repair by matching and modifying vertices, edges, and faces that ought to be merged in order to maintain the topology of a boundary representation. Because this is not always possible, more drastic repair procedures use triangulations to fill holes and gaps between adjacent faces [8, 7, 13] and/or allow substantial changes in topology or combinatorial structure for the sake of repair. A more successful of these approaches is the linear polyhedron repair method based on space decomposition into convex cells by planes associated with polyhedron's faces[34]. A valid boundary representation model is generated by re-evaluating the boundary of the union of all solid cells that are judged (heuristically) to classify inside the solid.

To summarize, all known repair techniques have limitations and provide few guarantees. They also may be computationally expensive. For example, it is known that optimal matching of vertices and/or edges for repair is NP-hard [8, 49]. But by far **the biggest problem with geometric healing algorithms is that they alter the original geometric data,** with possibly unpredictable and dangerous consequences. If the repaired data is translated into yet another receiving system, it may become invalid and in need of healing again; if the repaired data is translated back into the original sending system, it may or may not be valid again. In both cases, the geometric models before and after healing will never be the same.

The formal model of validity advocated in this paper is consistent with the overall philosophy of tolerant modeling, but it **does not seek or requires existence of exact r-set.** Under this model, we will also see in section 4.2 that some popular tolerant modeling algorithms are *not sufficient,* and that typical repair algorithms are *not necessary* for maintaining the solidity of boundary representations.

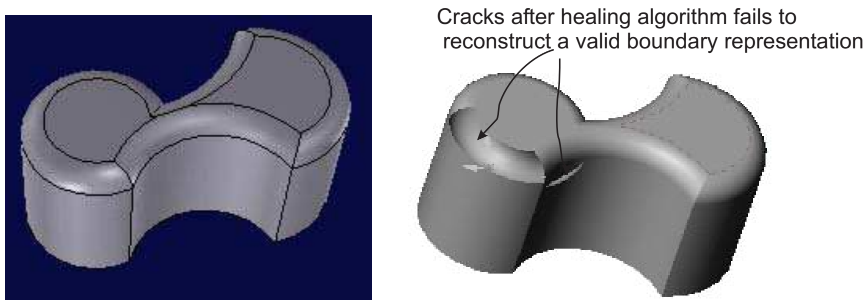
2 Formulation of Data Translation

2.1 Motivating Examples

Many references [19, 24, 16, 30, 37, 15] have illustrated various data translation problems. We will not attempt to add to the long list of well known difficulties, but rather consider few carefully chosen but real examples that provide important insights into the nature and intrinsic sources of the general translation problem. The choice of commercial systems in the following examples is not important, because the problems are generic. The described difficulties are representative of the current state of the art, and do *not* indicate inferiority of the any specific systems.

Example 1 The first example illustrates the well-known fact that even minor changes in geometric representation may invalidate the model, causing irreparable difficulties in data translation. In this case, the model shown in Figure 2(a) is created in SolidWorks and saved in the STEP (STandard for the Exchange of Product model data) neutral data exchange file format [1]. Then the STEP model is reloaded into SolidWorks, but was found to be invalid. The built-in healing algorithm attempted but was unable to recover a valid solid, generating instead the model shown in Figure 2(b).

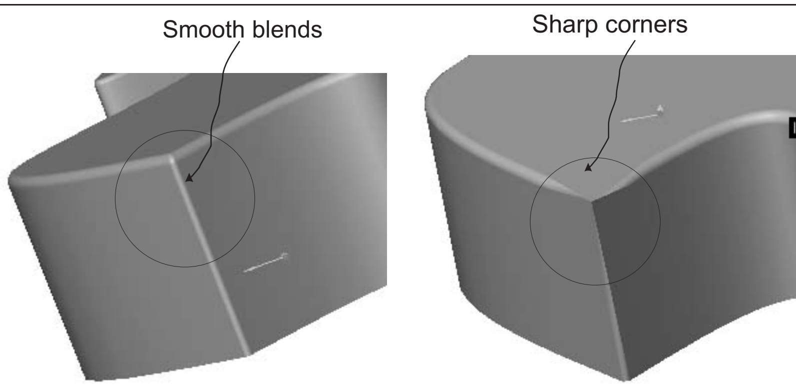
The above situation is common when geometric representations are archived in another non-native format. For example, saving the same model in ACIS format instead of STEP leads to similar difficulties. This double data translation corresponds to a situation in Figure 1 where no new errors are introduced in the evaluation algorithm by the receiving system (because it is the same as the sending system). The problem arises because primitives in the boundary representation – in this case, filleting surfaces and intersection spline curves in the original model – are mapped approximately into the STEP format by the translator.[4]



(a) (b)

Figure 2: Even minor changes in geometric primitives during translation may invalidate the model: (a) original model; (b) a failed attempt to repair the translated model.

Example 2 The second example (Figure 3(a)) is intended to show that even when geometric healing is successful in repairing the received model, the result may not be always acceptable. The double translation procedure is identical to the first example, except in this case, the healing algorithm is successful and generates the model shown in Figure 3(b). The smooth blends near the corner have been replaced by sharp corners in the translated model; such drastic changes are not acceptable for engineering applications where blend radius is an important parameter.



(a) (b)

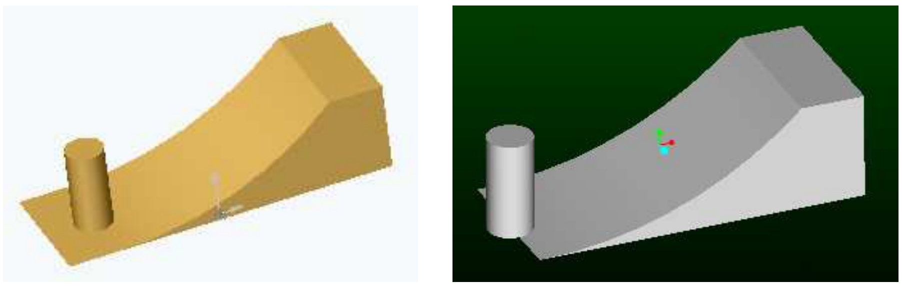
Figure 3: Healing algorithms may drastically change important geometric properties: (a) the original model with smooth blends; (b) an automatically repaired model with sharp corners.

Example 3 The third example shows that differences in precision of evaluating algorithms is another key ingredient of the translation difficulties, even when the changes in geometric representations are negligible. The solid model in Figure 4(a) was created in SolidWorks using only planar and cylindrical primitives with integer and fixed-precision coordinates. The dimensions of the model range from 0.001 mm (the minimum thickness of the part) to 1000 mm (the length of the part). The model is translated into Pro-Engineer through the STEP format, and both formats support exact representation of the original primitives. Therefore, it is reasonable to assume that the changes in geometric representation during the translation process remain negligible. Figure 4(b) shows the translated model after it is

[4]Similar translation problems are common whenever tangent surfaces are approximated in the course of translation.

evaluated in Pro-Engineer. It is certainly a valid solid, but with a drastically different shape that is not likely to be consistent with the intended use of the original solid.

This last example demonstrates clearly that a geometric representation alone does not uniquely define a set of points. Rather, the set of points, and therefore all its properties, are also determined by the properties (in particular, precision) of the evaluation algorithm. In this case, Solidworks relies on incidence testing algorithms with a default tolerance of 10E-6 mm, while Pro-Engineer uses a relative tolerance of 10E-6 times the maximum size of the bounding box of the model measured in meters. The latter effectively determines the smallest feature size to be 10E-6 meters, matching the minimum thickness of the model in Figure 4(a). The evaluation algorithm includes the process of merging what Pro-Engineer now considers coincident geometric entities, and results in the 'repaired' model shown in Figure 4(b).



(a) (b)

Figure 4: Evaluating the same geometric data with different precisions results in inconsistent solids: (a) the original model with small feature in the sending system; (b) the received model with small feature removed after healing.

2.2 Anatomy of Translation

Conceptually, every geometric translation procedure involves three ingredients: primitive mapping, rule mapping, and possibly modified evaluation algorithms. This view is reflected in Figure 1, is born out by the examples, and includes classical solid representation conversion problems. For example, when boundary representation is evaluated from Constructive Solid Geometry (CSG), primitive halfspaces in CSG are mapped into surfaces, curves, and vertices of the boundary representation; regularized set operations are replaced by combinatorial topological rules forming faces, edges, loops, and shells; and the CSG point membership classification (PMC) algorithm is replaced by the corresponding PMC on b-reps. Not all of the ingredients need to play a role in each translation problem. In each of the above translation Examples 1 and 2, the rule mapping is identity and the evaluation algorithm is unchanged. In Example 3, geometric primitives remain the same, but the evaluation algorithm is modified.

Primitive mapping is often accompanied by approximations that introduce data errors. The most common change in an evaluation algorithm corresponds to change in precision of point membership classification tests.[5] Under most scenarios, the transformation of rules is a matter of logic and semantics and generally does not introduce numerical errors. Based on these observations, our approach to the data translation problem recognizes explicitly the two sources of errors: uncertainty of geometric data and precision of PMC algorithm, while ignoring any issues associated with rule mappings.[6]

2.3 Validity and Consistency of Translation

From the above examples, we observe that there are two types of problems in data translation: *validity* and *consistency.* In Example 1, a solid that is deemed valid in the sending system is judged invalid in the receiving system. Examples 2 and 3 show that valid translations may not always produce expected solids that are consistent with intended use,

[5] It can be argued that all geometric algorithms sooner or later reduce to a finite number of point membership tests[47].

[6] This is not to say that rule mapping is a trivial issue; for example, the outstanding issue of persistent naming continues to undermine ability to translate and exchange parametric representations of solids.

particularly when healing algorithms alter the original geometric model. Furthermore, the classical notions of validity and consistency of solid models as defined by Requicha in [40] are not sufficient to formulate the translation problem. Under the classical definition, the geometric models in sending and receiving systems cannot be consistent in the presence of *any* approximations in geometric primitives. On the other hand, Example 3 clearly shows that validity of a geometric model is not absolute but is relative to a particular evaluation procedure used by a system.

In order to properly account for the roles of geometric approximations and algorithms that are specific to individual systems (recall Figure 1), we will say that *a* **translation is valid** *if the original geometric model is valid in the sending system and the translated model is valid in the receiving system.* Our definition of translation validity is axiomatic in the sense that it does not depend on specific interpretations of validity criteria. This implies that even different interpretations of model validity are permitted. For example, if a solid boundary representation is translated into a surface model, it may still be valid in a receiving system that performs surface visualization and/or area computations. Furthermore, this definition recognizes that model validity depends not only on the geometric model but also on the specific evaluation procedure used by the system.

Consistency of translation may be defined in the same spirit. We will say that a *valid* **translation is consistent** *if the original model and the translated model are indistinguishable under a specific comparison criteria.* Once again, this definition is axiomatic; it recognizes explicitly that the notion of consistency depends on a comparison criteria which may involve a variety of geometric (e.g., volume, area, minimum feature size, distance), topological (e.g., homeomorphism, homotopy, combinatorial), and other computable measures. It is logical that the notion of consistency applies only to valid translations, because at the very least, the translated models must be valid for the purpose of evaluating the specified comparison criteria. For example, if we want to compute the difference between volumes of the original model and the translated model, we need to know first that volume computations are supported by the corresponding systems.

The above definitions suggest an approach for formulating and solving geometric translation problem. Since validity is a necessary condition for consistency, it must be established first. Accordingly, the remainder of this paper deals with validity of translation of solid models. This in turn requires formulating and evaluating solidity of geometric models in the respective systems in the presence of errors and approximations.

We chose not to explore translation consistency in this paper for several reasons. First and foremost, consistency is clearly an application specific notion. For example, in certain applications (for example packaging), it may be perfectly acceptable to translate a complex solid into its convex hull or a containing ball, while this clearly would not be acceptable for purpose of process planning or detailed design. Thus, it would be counterproductive to put the issue of consistency before the issue of validity, or worse yet to use an arbitrary assumed concept of consistency in place of validity – a common practice in many geometry healing and repair applications.

Arguably, validity without consistency does not solve the whole problem, but without proper notion of validity, we have no hope for verifiable solutions to consistency problems. A proper notion of solidity should include the classical theory as a special exact case (thus supporting exact computational models), support currently used ad hoc solutions (such as using the same modeling kernel throughout the translation process or adjusting tolerances to increase model robustness), and allow for trivial or drastic translations such as in the above examples. At the same time, since the representation rules are often preserved under translation (thereby eliminating from consideration trivial translation cases), the issue of validity in the presence of errors and variable precision is far from trivial.

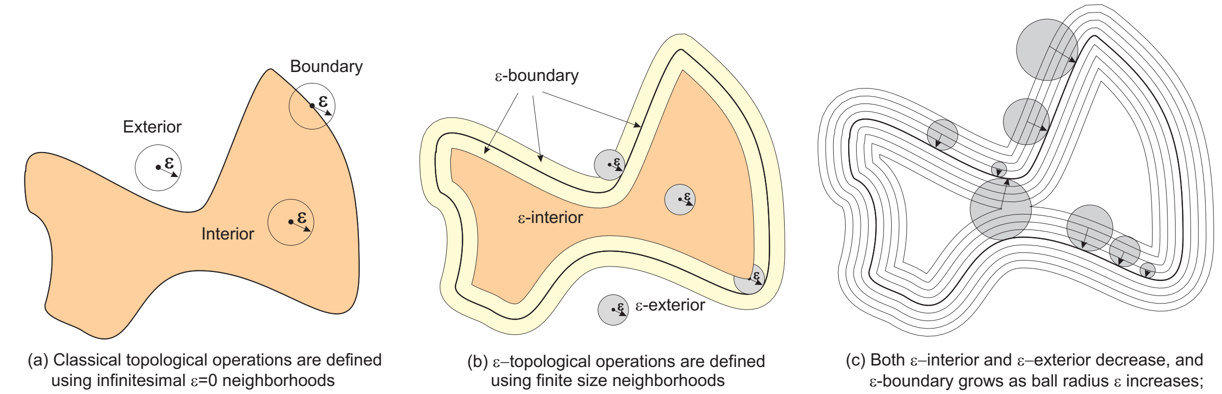
3 Solidity with Finite Neighborhoods

In this section, we propose a theory that captures the notions of inexact data and imprecise algorithms. The theory relies on finite ε-size neighborhoods to define ε-topological operations, which are then used to formulate the concept of ε-regularity and, finally, the notion of ε-solidity. The theory contains the classical theory of solid modeling as a special (though unrealistic) case. We will consider applications of the theory to data translation in Section 4.

3.1 Epsilon-Topological Operations

If we accept that both geometric data and algorithms are inherently imprecise, we have no choice but to reconsider the very foundations of the solid modeling theory. Traditionally, solidity is formulated in terms of the classical topological concepts and operations of closure k, interior i, and boundary ∂ that are interpreted in terms of infinitesimal size neighborhoods. In contrast, inaccuracy of data and finite resolution of the algorithms imply that the neighborhoods of every point may be represented only up to some finite size. This in turn requires redefining the usual topological operations.

Specifically, we propose epsilon-topological counterparts of the classical topological operations: *ε-closure* k_ε, *ε-interior* i_ε, and *ε-boundary* ∂_ε, where ε is a non-negative real number. Intuitively, ε corresponds to the maximum algorithm precision and/or maximal data precision. In the following definitions, we refer to a metric space as (W, d), where W is a non-empty set and d is a suitable distance function. In a metric space, $B(x, r)$ denotes the open ball about point x of radius r.



(a) (b) (c)

Figure 5: Classical topological operations and the corresponding ε-topological operations

Definition 3.1. Given a subset X of a metric space, a point x is said to be in the *ε-closure* of X, denoted $k_\varepsilon(X)$, if for every $r > \varepsilon$, $B(x, r) \cap X \neq \emptyset$, where ε is a non-negative real number.

Definition 3.2. Given a subset X of a metric space, a point x is said to be in the *ε-interior* of X, denoted $i_\varepsilon(X)$, if there exits an open ball about x of radius $r > \varepsilon$ such that a $B(x, r) \subseteq X$, where ε is a non-negative real number.

Definition 3.3. Given a subset X of a metric space, a point x is said to be in the *ε-boundary* of X, denoted $\partial_\varepsilon(X)$, if x is in both the $k_\varepsilon(X)$ and the $k_\varepsilon c(X)$, where $c(X)$ denotes the complement of a set.

The above ε-topological operations are illustrated in Figure 5. For $\varepsilon = 0$, these operations correspond to the usual classical topological operations, and in this sense, they are generalizations of the corresponding definitions in the general (point-set) topology [31, 32]. But for $\varepsilon > 0$, additional points are added or subtracted from closure or interior respectively, and the boundary of the set is "thickened" by the ball of radius ε. Notice that operations k_ε and i_ε define sets that are closed and open in the usual metric topology with $\varepsilon = 0$. It is not difficult to show that the ε-topological operations inherit and preserve many of the classical properties, as illustrated by the theorems below. In what follows, $c(X)$ denotes the complement of set X.

Theorem 3.4. *A point x is in the complement of $k_\varepsilon(X)$ if and only if there is an open ball about x of radius $r > \varepsilon$ such that $B(x, r) \cap X = \emptyset$.*

Proof. If $B(x, r) \cap X = \emptyset$ with $r > \varepsilon$, then it is also true that $B(x, r) \cap k_\varepsilon(X) = \emptyset$ with $r > 0$. Therefore, x is not in $k_\varepsilon(X)$. Conversely, if x is in in the complement of $k_\varepsilon(X)$, there is a $B(x, r) \cap k_\varepsilon(X) = \emptyset$ with $r > 0$, and therefore $B(x, r) \cap X = \emptyset$ with $r > \varepsilon$. □

Theorem 3.5. $i_\varepsilon(X) = ck_\varepsilon c(X)$

Proof. If x is a point in $ck_\varepsilon c(X)$, then there is a $B(x, r)$ of $r > \varepsilon$, $B(x, r) \cap c(X) = \emptyset$, or to say $B(x, r) \in X$, so from Definition 3.2, x is a point in $i_\varepsilon(X)$. Conversely, if x is a point in $i_\varepsilon(X)$, then there is a $B(x, r)$ of $r > \varepsilon$, $B(x, r) \in X$, so $B(x, r) \cap c(X) = \emptyset$, so from Theorem 3.4, x is a point in $ck_\varepsilon c(X)$. □

Corollary 3.6. $ci_\varepsilon(X) = k_\varepsilon c(X)$, $k_\varepsilon(X) = ci_\varepsilon c(X)$.

Similarly, it can be shown that many (but not all[7]) classical theorems are preserved under these generalized topological operations [17]. From the above definitions, a point cannot be in both ∂_ε and i_ε. Thus, every set $k_\varepsilon(X)$ is partitioned into its ε-interior and a 'thickened' boundary as $k_\varepsilon(X) = \partial_\varepsilon(X) \cup i_\varepsilon(X)$. It follows immediately that for any set X,

$$i_\varepsilon(X) \subseteq X \subseteq k_\varepsilon(X). \tag{1}$$

The set of points that are *not* the $k_\varepsilon(X)$ is called *ε-exterior* of set X and will be denoted $e_\varepsilon(X)$. It can be also defined directly following Theorem 3.4 above. Since

$$e_\varepsilon(X) \cup \partial_\varepsilon(X) \cup i_\varepsilon(X) = W, \tag{2}$$

we can say that any set $X \subseteq W$ induces a partition of the space W under the ε-topological operations.

Consider what happens to the induced partition of space W under different values of ε (Figure 5(c)). For a given set X, the larger value of ε will result in a shrunk ε-interior $i_\varepsilon(X)$ and a grown ε-closure $k_\varepsilon(X)$. As the result, the ε-boundary will thicken further as well. We can summarize this concisely by the theorem whose proof follows directly from the above definitions.

Theorem 3.7. *For a given subset X of a metric space, and non-negative numbers ε_1 and ε_2, if $\varepsilon_1 > \varepsilon_2$, then*

$$i_{\varepsilon_1}(X) \subset i_{\varepsilon_2}(X); \qquad e_{\varepsilon_1}(X) \subset e_{\varepsilon_2}(X); \qquad \partial_{\varepsilon_1}(X) \supset \partial_{\varepsilon_2}(X) \tag{3}$$

The Theorem 3.7 implies that i_ε and e_ε are monotonically decreasing functions of ε: as ε decreases, points can only be added to the interior $i_\varepsilon(X)$ and exterior $e_\varepsilon(X)$, while they are removed from the shrinking boundary $\partial_\varepsilon(X)$. As ε approaches 0, $e_\varepsilon, i_\varepsilon, \partial_\varepsilon$ approach the classical exact sets of exterior, interior, and boundary respectively.

3.2 ε-Regular Sets

The classical models of solidity postulate that a solid is a set of points with dimensionally homogeneous interior and well defined boundary. This notion can be captured by requiring every valid computer representation correspond to at least one well defined *regular* set X:

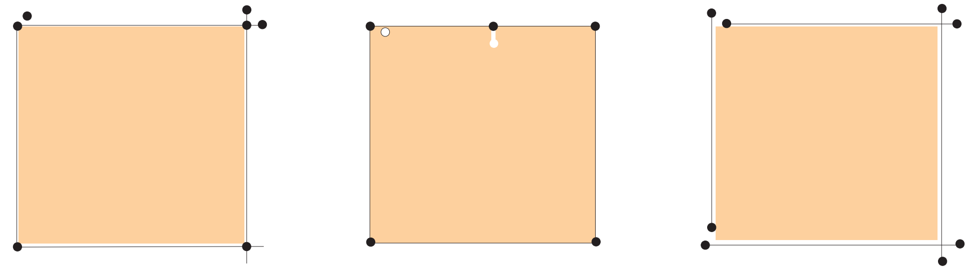
- Closed regular[40], in which case $X = ki(X)$, or
- Open regular[5], in which case $X = ik(X)$

Both models correspond to dimensionally homogeneous (without cracks or dangling pieces) sets with tight boundaries, the main difference being that closed regular sets include their boundaries while open regular do not. A common characterization of solidity in both models is that a neighborhood of every boundary point contains points in the solid's interior as well as points in its exterior.

As we argued above, identifying a regular set in the presence of errors is not always feasible, practical, or even desirable. Representation of X is usually generated from user inputs, numerical computations, approximations, and physical samplings – all of which contain some *errors*. These data errors in representation of X usually violate the validity conditions (numerical, topological, combinatorial) implied by the properties of regular sets. For example, Figures 6 shows typical 'invalid' 2-dimensional boundary representations of a simple rectangle; as with most boundary representations, the vertices, edges, and faces represent geometric information only approximately and redundantly. Depending on how the boundary is constructed, it also may include small imperfections: isolated and dangling pieces, voids, gaps. Applying the notions of (exact) closure, interior, and boundary to such representations and requiring regularity does not really make sense, because strictly speaking, all such representations are invalid. Hence, the key question is: *What is a proper model of solidity in terms of ε-topological operations* that would tolerate errors of size less than ε, such as those shown in Figure 6? Below, we answer this question by proposing notions of ε-regularity and ε-solidity that subsume the corresponding classical notions.

Intuitively, it should be apparent that a proper definition of regularity must not eliminate but *tolerate* errors and imperfections near the solid's boundary. The definition should be consistent with the classical notion of regular sets, and in fact the generalization is relatively straightforward. We *postulate* that a *known* set of X should be deemed ε-regular if *an ε-neighborhood of every boundary point $p \in \partial_0$ contains points in the solid's interior i_0 and points in*

[7]Notably, the classical property $k(k(X)) = k(X)$ does not hold for k_ε operation when $\varepsilon > 0$.



(a) Dangling edges or isolated points (b) Small cracks and voids (c) Misaligned or redundant vertices and edges

Figure 6: A theory of ε-solidity must tolerate imperfections of size less than ε near the theoretical boundary of a solid.

solid's exterior e_0. The challenge is to formulate this postulate in terms of the ε-topological operations in a manner that is consistent with the classical notions of regularity.

Let us first consider what this means when $\varepsilon = 0$. By definition, $i_0(X)$ is the largest open set contained in X, and $k_0(X)$ is the smallest closed set that contains X, which means that for any set X

$$i_0(X) \subseteq X \subseteq k_0(X) \tag{4}$$

Neither one of these bounding sets are guaranteed to be homogeneous in dimension: $i_0(X)$ may contain void and cracks, while $k_0(X)$ may include dangling pieces and isolated points. The homogenization (or regularization) is achieved by a second topological operation that grows the lower and shrinks the upper bounding sets respectively: $k_0 i_0(X)$ is the smallest closed regular set containing $i_0(X)$, and $i_0 k_0(X)$ is the largest open regular set contained in $k_0 X$. The regularization effectively reverses the set relationship (4) to

$$i_0 k_0(X) \subseteq X \subseteq k_0 i_0(X) \tag{5}$$

It should be clear that any open or closed regular set X satisfies the inequality (5) whose essence is to combine the two classical definitions into a single definition of an 0-regular set. But other sets satisfy the above inequality as well, because it allows for imperfections (missing portions or isolated points) in the boundary $\partial_0 X$. For example, a unit sphere with half of its boundary missing would satisfy (5) and be considered 0-regular.

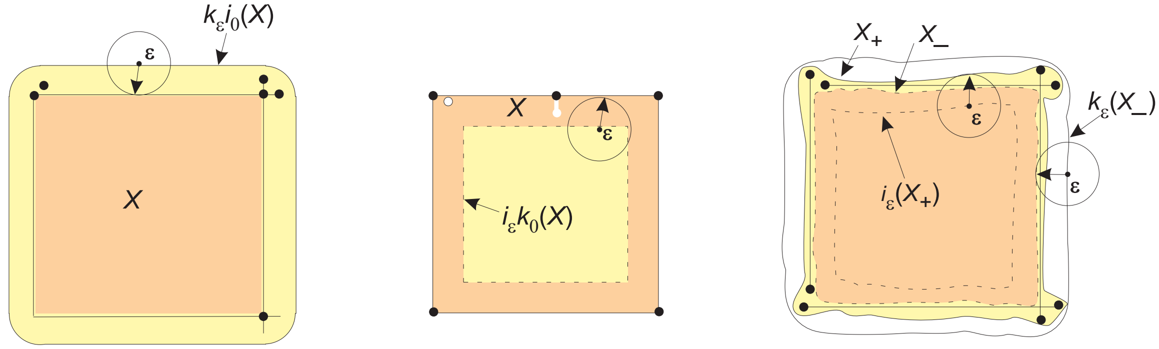
However, the two homogeneous bounds (5) are so tight that they are not realistic, because no set X satisfying (5) may contain even minor imperfections in its interior or exterior, such as those shown in Figure 6. To say that imperfections are tolerated in the interior of the set X *within distance* ε from the boundary amounts to a statement that X contains $i_\varepsilon k_0(X)$; similarly, when $k_\varepsilon i_0(X)$ contains the set X, exterior imperfections within ε from the boundary are tolerated. This motivates our first attempt at definition of ε-regularity for a *known* set X.

Definition 3.8. A subset X of a metric space is ε*-regular* if, for a given non-negative real number ε,

$$i_\varepsilon k_0(X) \subseteq X \subseteq k_\varepsilon i_0(X). \tag{6}$$

Informally, the above definition assumes that we can determine the closure and interior of X exactly, but we allow errors within distance ε near the boundary. The concept is illustrated in Figure 7 showing that imperfections within ε of the boundary in Figure 6 are "covered" by either growing $i_0(X)$ by k_ε as in Figure 7(a), or by shrinking $k_0(X)$ by i_ε as in Figure 7(b). Thus, in contrast to the classical definitions, under the proposed notion of ε-regularity, these representations of rectangles are considered to represent ε-regular sets.

By definition 3.8, ε-regularity of a set depends on the size of chosen ε. Classical regular sets (open and closed) clearly satisfy the definition in the special case when $\varepsilon = 0$, with one side of (3.8) becoming an equality. Inequality is crucial in the general case, because it corresponds to the notion of tolerant modeling near the boundary of the set. By definition, increasing values of ε corresponds to shrinking $i_\varepsilon k_0(X)$ and expanding $k_\varepsilon i_0(X)$. Thus, if X is ε-regular, it is also ε_1-regular for any $\varepsilon_1 \geq \varepsilon$. We observe that every bounded set (or rather every set containing bounded 'imperfections') with non-empty interior is ε-regular for some sufficiently large value of ε. Normally, we are interested in the smallest value of ε for which X is ε-regular.



(a) Growing $i_0(X)$ to cover dangling edges and isolated points

(b) Shrinking $k_0(X)$ to cover small cracks and voids

(c) An interval [X_, X_+] coverers misaligned or redundant vertices and edges by $i_\varepsilon(X_+)$ and k_ε(X_)

Figure 7: ε-regular sets and intervals tolerate imperfections of size less than ε near the set boundary.

3.3 ε-Regular Intervals

We now consider more realistic situations where we may not be able to compute the interior $i_0(X)$ and closure $k_0(X)$ of a set X exactly, but only some bounding sets X_- and X_+ can be determined – either as input or as a result of another approximate computation. In what follows, we will refer to the bounding set X_- as *inner*, and X_+ as *outer*. An example of inner and outer bounding sets is shown in Figure 7(c). The interior $i_0(X)$ and the closure $k_0(X)$ are the tightest bounds computable for any given set X. The inner and outer sets form a *set interval*, $[X_-, X_+]$ — *the class of sets* $\{X\}$ such that $X_- \subseteq X \subseteq X_+$. Without loss of generality, we will assume that the inner X_- of any set interval is open and the outer X_+ is closed. Then the definition 3.8 generalizes in a straightforward fashion by replacing $k_0(X)$ and $i_0(X)$ respectively with arbitrary outer X_+ and inner X_-:

Definition 3.9. A set interval $[X_-, X_+]$ is ε-*regular* if, for a given non-negative real number ε,

$$i_\varepsilon(X_+) \subseteq X_- \subset X_+ \subseteq k_\varepsilon(X_-) \tag{7}$$

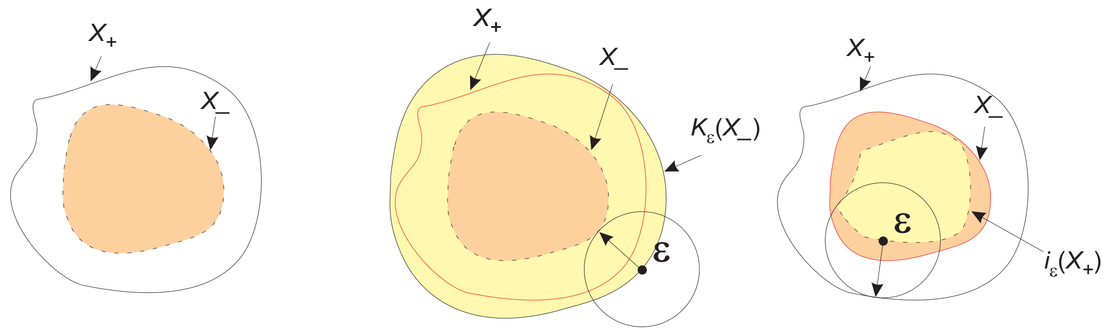
Technically, the test for ε-regularity of an interval depends on two separate conditions illustrated in Figure 8: $i_\varepsilon(X_+) \subseteq X_-$ requires that when outer X_+ is shrunk by a ball of size ε, it fits inside the inner; similarly $X_+ \subseteq k_\varepsilon(X_-)$ requires that the inner X_- grown by a ball of size ε contains the outer X_+. For example, the rectangle in Figure 7(c) is not regular in the classical sense, but is ε-regular as an interval $[X_-, X_+]$. Once again, if the interval is ε-regular for any particular value of ε, then it must also be ε-regular for any greater value of ε, but not necessarily for the smaller.

The definition 3.9 for a set interval is written in the same form as definition 3.8 for a set instance, in order to emphasize their common structure. In fact, it is easy to show that Definition 3.8 is a special case of Definition 3.9, by recalling that any set X is contained in the interval $[i_0(X), k_0(X)]$.

Theorem 3.10. *A set X is ε-regular iff the interval $[i_0(X), k_0(X)]$ is ε-regular.*

Proof. From Definition 3.8, we have $k_\varepsilon i_0(X) \supseteq k_0(X)$, and $i_0(X) \supseteq i_\varepsilon k_0(X)$. Thus, let $X_- = i_0(X)$ and $X_+ = k_0(X)$. For the interval $[X_-, X_+]$, we have $k_\varepsilon X_- \supseteq X_+$, and $X_- \supseteq i_\varepsilon X_+$, which is an ε-regular interval. Conversely, if $[i_0(X), k_0(X)]$ is ε-regular, then from Definition 3.9, $k_\varepsilon i_0(X) \supseteq k_0(X)$, $i_0(X) \supseteq i_\varepsilon k_0(X)$. Thus, X is ε-regular. □

In other words, we really need only one definition 3.9 of ε-regular interval, because it subsumes definition 3.8 of an ε-regular set. Henceforth, it should be understood that a term 'ε-regular interval' also applies to ε-regular set instances. Furthermore, it is easy to see that every set instance X in a ε-regular interval $[X_-, X_+]$ is ε-regular. This is reasonable and should be expected, since every such set interval represents an equivalence class of sets that are not distinguishable beyond the inner and outer bounds of the interval. In fact, if we define a *subinterval* $[Y_-, Y_+]$ of interval $[X_-, X_+]$ with $Y_- \supseteq X_-$ and $Y_+ \subseteq X_-$, we can make an even stronger claim:



(a) A given set interval $[X\_, X_+]$

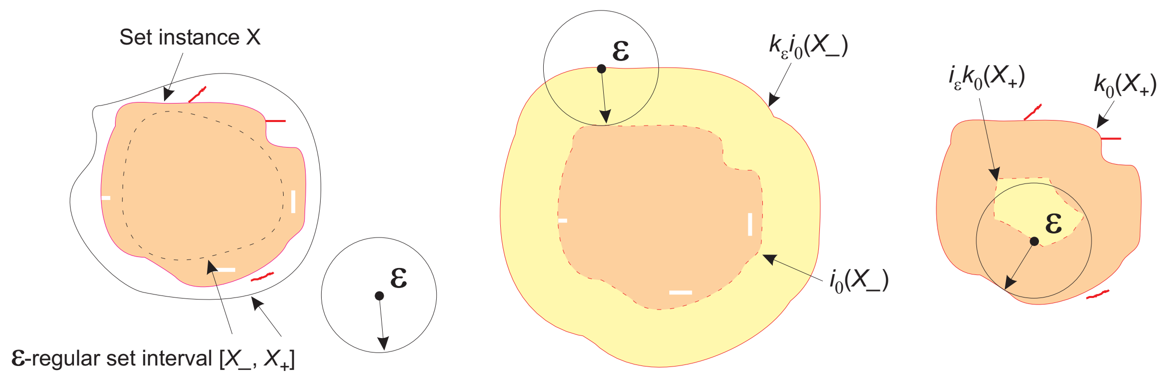
(b) The set interval is ε-regular since $X_+ \subseteq k_\varepsilon(X\_)$ and $i_\varepsilon(X_+) \subseteq X\_$

Figure 8: A set interval becomes ε-regular if the ε is big enough.

Theorem 3.11. *Any subinterval $[Y_-, Y_+]$ of an ε-regular interval $[X_-, X_+]$ is also ε-regular.*

Proof. By definition, X_-, Y_- are open, and X_+, Y_+ are closed sets, with $Y_- \supseteq X_-$ and $Y_+ \subseteq X_+$. Since $k_\varepsilon(X_-) \supseteq X_+$, then $k_\varepsilon(Y_-) \supseteq k_\varepsilon(X_-) \supseteq X_+ \supseteq Y_+$; similarly, since $X_- \supseteq i_\varepsilon(X_+)$, then $Y_- \supseteq X_- \supseteq i_\varepsilon(X_+) \supseteq i_\varepsilon(Y_+)$. Thus, $[Y_-, Y_+]$ is ε-regular. □

This result is of paramount *practical* significance, because it allows to verify regularity of an interval $[Y_-, Y_+]$ even when the interval itself is not computable by testing a larger containing interval $[X_-, X_+]$ that is computable. In particular, Figure 9 shows that any set X contained in an ε-regular interval must be ε-regular. This statement is conservative in a sense that X may be ε-regular with even a smaller value of ε.



(a) An ε-regular set interval has a set instance X with dangling pieces and inner cracks

(b) Set instance X is ε-regular as $X \subseteq k_\varepsilon i_0(X\_)$ and $i_\varepsilon k_0(X_+) \subseteq X$

Figure 9: Any set contained in an ε-regular interval is ε-regular with the same or smaller value of ε.

3.4 ε-Solidity

The ε-regularity extends the notion of solid homogeneity to set intervals and, as such, is a necessary condition for the notion of ε-solidity. Note however, that the inner X_- and the outer X_+ of an ε-regular interval do not need to be dimensionally homogeneous sets. It is also customary to require that every solid is bounded and has a non-empty interior. These statements translate respectively into additional constraints on set intervals requiring that the inner X_- is non-empty and the outer X_+ is bounded.

Definition 3.12. An ε-*solid* is an ε-regular set interval $[X_-, X_+]$ with non-empty X_- and bounded X_+.

Given any computer representation of a set, its interior and closure can be computed only within some precision δ. In this case, $i_\delta(X) \subseteq X \subseteq k_\delta(X)$, but since X is really not known, it is more reasonable to consider interval $[i_\delta(X), k_\delta(X)]$ in place of the set. If $i_\delta(X)$ is not empty and $k_\delta(X)$ is bounded, this interval is an ε-solid if it is ε-regular, for a particular choice of ε. Following definition 3.9, the later requires satisfaction of two conditions:

$$i_\varepsilon k_\delta(X) \subseteq i_\delta(X), \qquad \text{and} \qquad k_\delta(X) \subseteq k_\varepsilon i_\delta(X). \tag{8}$$

The choice of ε is critical. Smaller ε is preferable because it corresponds to a more accurate representation, but if we choose $\varepsilon < \delta$, then the interval *cannot* be regular because $k_\varepsilon i_\delta(X) \subseteq X \subseteq i_\varepsilon k_\delta(X)$. The regularity conditions (8) can only be satisfied starting with some $\varepsilon \geq \delta$. At this point, we would have established that the interval $[X_-, X_+]$ is indeed an ε-solid according to the definition 3.12. Furthermore, the following theorem is an immediate corollary to Theorem 3.11 and implies that the set X itself is also an ε-solid.

Theorem 3.13. *Any subinterval $[Y_-, Y_+]$ of ε-solid $[X_-, X_+]$ is ε-solid.*

Proof. From Theorem 3.11, any subinterval in $[X_-, X_+]$ is ε-regular. Since the inner bound X_- is non-empty, so is the corresponding inner bound of the subinterval Y_-; the boundedness of the outer set X_+ implies the boundedness of Y_+. Thus the subinterval $[Y_-, Y_+]$ is an ε-solid. □

For example, if we can establish that any $[i_\delta(X), k_\delta(X)]$ is ε-solid, then our formulation implies that $[i_0(X), k_0(X)]$, and hence X itself, is also ε-solid – even though X may contain errors or may not be known. For a specific given value of $\varepsilon \geq \delta$, X may or may not be ε-solid. However, unless set X is unbounded or its interior $i_\delta(X)$ is empty, it is an ε-solid for *some* sufficiently large value of ε. An essentially identical procedure may be used to determine if an arbitrary interval $[X_-, X_+]$ is ε-solid.

The above notions of ε-regularity and ε-solidity recognize explicitly that the closure and interior of any set may be determined only up to some limited precision δ and may be tested for regularity only within some resolution $\varepsilon \geq \delta$. Thus, a given ε-solid X now may or may not be solid in the classical sense, depending on the specific values of δ and ε. This is reasonable, since in practice we are not likely to have exact representations and computations of X.

4 Validity and Translation of ε-Solid Representations

The above definitions and theorems explicitly acknowledge that topological properties of geometric data may be represented or computed only within some finite precision. The challenge is to build on these definitions, in order to formulate the notion of validity of solids and translations in the presence of errors and limited precision, and to develop practical algorithms that implement these definitions. Any credible solution to the geometric data translation problem must acknowledge fixed precision of algorithms that perform geometric evaluations and queries, as well as the fact that this precision varies from system to system.

4.1 Accuracy of Data and Precision of Algorithms

Representations of geometric objects are stored on a computer and queried by algorithms. Below we will distinguish between the *accuracy* of a geometric representation and the *precision* of geometric queries on this data. The accuracy of any point of the geometric data is measured by a λ-radius of the uncertainty ball about that point. If the data comes from the ideal source and is represented exactly, $\lambda = 0$; if the source of the data is only approximate or the data underwent an approximate conversion, $\lambda > 0$. Unfortunately, accuracy of geometric data is not commonly archived. We will argue below that recognizing explicit changes in accuracy of geometric data can help to classify and sometimes avoid common difficulties in the data translation problems.

Most geometric algorithms and queries on geometric data sooner or later reduce to a finite number of Point Membership Classification (PMC) tests[47]. For a given point p and a set X, PMC returns IN, ON or OUT depending on whether p belongs to the interior, boundary, or exterior of X respectively. The classical semantics of PMC in solid modeling assumes an ability to compute an arbitrary small neighborhood of point p in order to identify the interior, exterior and boundary of X. However, all practical implementations of PMC algorithms have finite precision and always rely on finite size neighborhoods implied by floating point roundoff errors and approximations. Even when a set X is represented exactly ($\lambda = 0$) using a finite collection of geometric primitives, PMC tests near the boundary of the primitives have limited precision, and return ON not only for the points on the boundary but also for the points that are near the boundary. Thus, when we say that a geometric representation can queried with precision δ, we specifically refer to precision of a PMC$_\delta$ test that is defined in terms of the ε-topological operations as:

$$\mathrm{PMC}_\delta(p, X) = \begin{cases} IN & \text{if } p \in i_\delta X \\ OUT & \text{if } p \in e_\delta X \\ ON & \text{if } p \in \partial_\delta X \end{cases}$$

In other words, PMC$_\delta$ is just an operational definition of the partition of space (2) induced by any set X under the δ-topological operations. It is a straightforward generalization of the classical PMC test that reflects the role of precision in practical implementations of the test. The classical PMC is just a special exact case of the above test with $\delta = 0$. Informally, PMC$_\delta$ classifies a point p as IN or OUT with respect to a given set X if it is farther then δ from the set boundary, otherwise the point is classified as ON. Since X itself is usually represented with some finite accuracy λ, PMC$_\delta$ test make sense only when its precision is greater than accuracy of the data i.e. under all conditions, we must have $\delta \geq \lambda$ — irrespectively of the semantics of the representation scheme.

The precision of PMC$_\delta$ is a major limiting factor in deciding whether a geometric data represents a valid ε-solid or not. Recall from section 3.4 that an interval $[i_\delta(X), k_\delta(X)]$ may be an ε-solid *only* when $\varepsilon \geq \delta$. It may be tempting to identify δ and ε and treat them as one constant, but the distinction is semantically important, as illustrated by the following examples.

Consider a standard query on a CSG representation [53]. Figures 10(a)-(c) show the classical regularization intersection operation that assumes exact representation of the primitive sets A and B and ability to compute arbitrarily small neighborhoods. In practice, however, depending on accuracy of CSG primitives and PMC$_\delta$ algorithm, a point may be classified with respect to each primitive only within some precision δ. The classifications from individual primitives are combined according to the regularized set operations (Figure 10(d)), but the regularized combination operation depends critically on explicit representation and analysis of the neighborhood of the point. As shown in Figure 10(e), the size of the neighborhood ε must be greater or equal to δ, or regularization may fail, producing results that are inconsistent with the semantics of the exact regularized set operations.

The distinction between δ and ε is even more pronounced for boundary representations. Consider the boundary representation of a rectangle in Figure 13(a). The faces (in this case, the line segments) may or may not be represented exactly (i.e. λ may or may not be 0), but their connectedness may be established only within precision $\delta \geq \lambda$. Depending on selected value of ε, the connected set itself may or may not be a boundary of ε-solid, but under all conditions we must have $\varepsilon \geq \delta$ (Figure 13(b)).

4.2 Validity and Repair of Representations

The ε-solid model is a generalization of the classical r-set solid model that requires a significant revision of the modeling paradigm proposed in [40], as shown in Figure 11. The extended mathematical *modeling space* is the space of all ε-solids, which includes all closed and open regular sets. For any given representation scheme, the *representation space* is the space of representations – pairs (R_λ, PMC$_\delta$), where R_λ is a geometric data represented with accuracy λ and PMC$_\delta$ is a point classification procedure with precision $\delta \geq \lambda$. The *representation scheme* itself is a mapping from the space of ε-solids to representation pairs (R_λ, PMC$_\delta$). In this updated modeling paradigm, the definition of representation *validity* reflects more accurately how solids are handled in modern modeling systems.

Definition 4.1. Representation (R_λ, PMC$_\delta$) is *valid* if application of PMC$_\delta$ on R_λ induces an ε-solid.

This notion of validity should support systematic development of provably correct algorithms and will allow to validate and verify existing and proposed methods. As a demonstration, it is easy to argue *based on empirical evidence and assumed theoretical foundations* that popular methods of “healing” geometric boundary representations are neither necessary nor sufficient to solve the data translation problem.

In classical solid modeling, validation of boundary representation requires proper connectivity of the vertices, edges, faces, and verification that shells of a boundary representation are two-manifolds [40, 26, 43]. Theoretically, the metric conditions of such a model are required to be exact, and when geometric carriers (curves, surfaces) improperly overlap or do not match “closely enough,” geometric healing techniques attempt to correct “invalid” solids by various surgical operations.

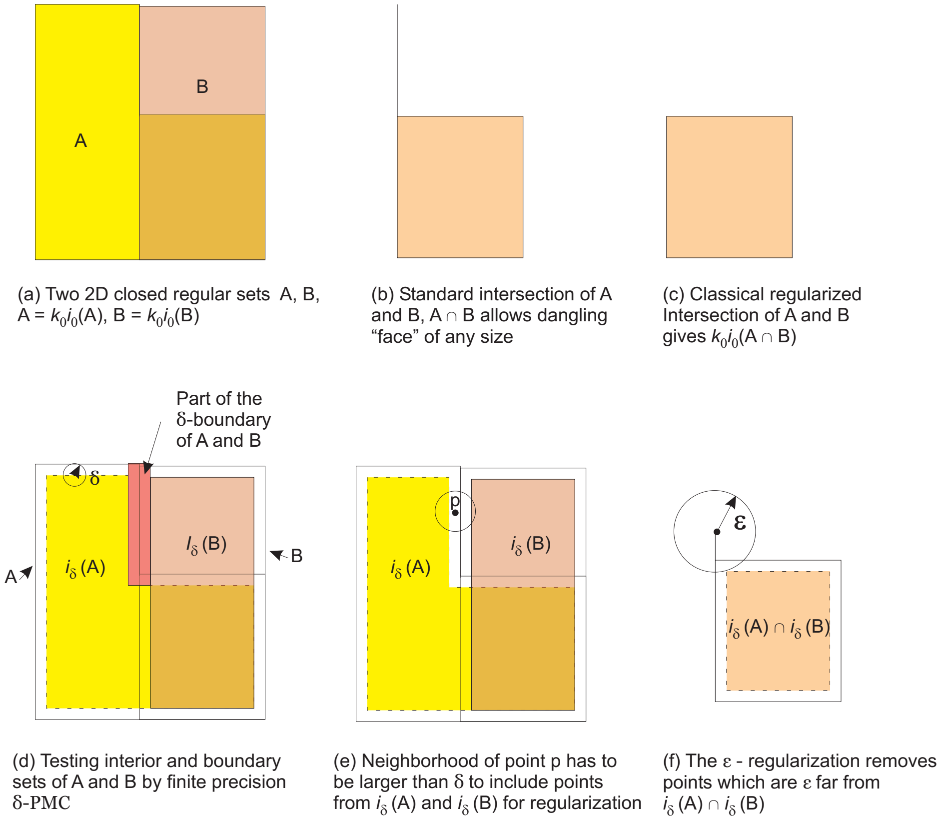


Figure 10: Comparison of the theoretically exact PMC on CSG representation (top row) and PMC$_\varepsilon$ recognizing limited precision of PMC$_\delta$ against individual primitives

Figure 12(a) shows a tessellated boundary representation of a circle. This is a valid polygon, but every vertex has a PMC tolerance zone of size δ, and the zones of adjacent vertices overlap as shown. One popular technique used for robust geometric computations merges the adjacent vertices whenever their zones overlap into a single vertex with a larger zone[45]. In our example, this process will result in a single vertex with a tolerance zone covering the whole polygon, suggesting that the object is not a valid polygon. But as shown in Figure 12(b), the polygon is bounded has a well-defined non-empty δ-interior and δ-closure. In other words, a suitable PMC$_\delta$ test on the representation gives an ε-regular set interval for some number ε. This example shows that *solid validation and healing procedures based solely on boundary representations are not sufficient.* Another familiar and simple example is shown in Figure 13(a). In this case, a typical boundary-based validation procedure would determine that the representation is invalid, because the adjacent edges don't intersect at common vertices. A healing procedure would usually be invoked to re-intersect the edges, in order to get more precise vertices. But it should be clear from the from Figure 13(b) and discussion in sections 3.4 and 4.1, that with suitable PMC$_\delta$ we can still induce the interior and exterior of the rectangle — even though the connectivity of adjacent vertices and edges is not guaranteed. In other words, *boundary-based healing procedure are not necessary.*

4.3 Maintaining ε-solids in Data Translation

The proposed formulation of geometric data translation problem relies on validity of the geometric model respectively in the sending and the receiving systems, see Section 2. Thus, we will *assume* that the geometric model is valid in the sending system, and ask: *Under what conditions the translated model is valid in the receiving system?* The term "valid" can now be interpreted to mean that "geometric model with accuracy λ is an ε-solid with respect to a suitable PMC$_\delta$ algorithm with precision δ". Let λ, δ, ε be respectively data accuracy, algorithm precision, and solidity measure

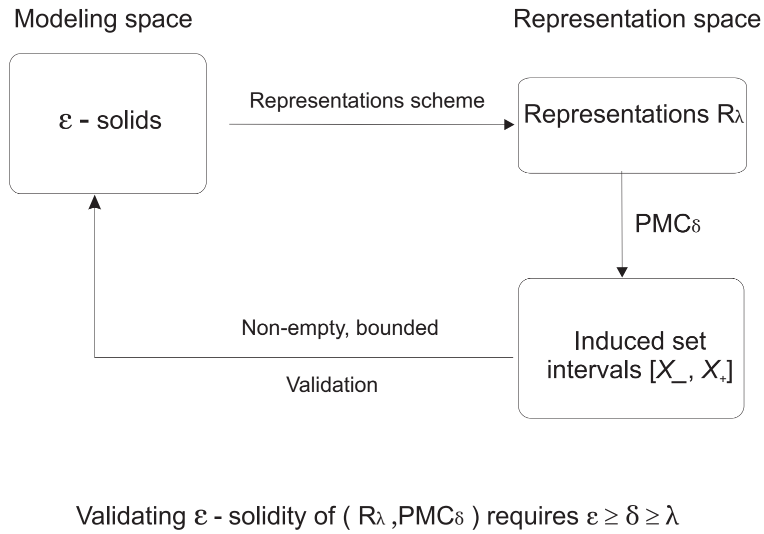


Figure 11: Modeling and representation spaces for ε-solids with limited data accuracy λ and algorithm precision δ

in the sending system; and let λ', δ', ε' be the corresponding quantities in the receiving system. There are four possible types of data translation, depending on possible changes to data accuracy λ and algorithm precision δ:

Accuracy is fixed $\lambda = \lambda'$, precision is fixed $\delta = \delta'$. The sending system and the receiving system have the same accuracy of geometric data and precision of evaluation algorithms, and there are no approximations in translation. Then, if the model X was ε-solid in the sending system, it will remain ε-solid in the receiving system.

Accuracy is changed $\lambda \neq \lambda'$, precision is fixed $\delta = \delta'$. Both systems evaluate the model with the same precision, but geometric model in the receiving system has a different accuracy λ'. This happens, for example, when a model is archived in a neutral format and subsequently reloaded into the native system (recall Example 1 in section 2.1), or when curved surface model is tesselated into an approximate polyhedron model. The received model may or may not be valid under the same precision $\delta' = \delta$. Specifically, when the received model is less accurate with an increased value of λ', the precision δ' of the PMC$_{\delta'}$ procedure may also need to be increased to maintain $\delta' \geq \lambda'$ in the receiving system. If the received model has smaller λ' than the original λ value of the sending model, δ' automatically satisfies $\delta' > \lambda'$ in the receiving system. Assuming that condition $\delta' \geq \lambda'$ is satisfied, the receiving system induces a different set interval $[X'_-, X'_+]$ from the original $[X_-, X_+]$ in the sending system. If inner X'_- becomes empty or outer X'_+ becomes unbounded, then the received model can not be a valid solid. However, in most practical situations, the received model approximates closely to the sending model, and the conditions of non-empty inner and bounded outer are usually maintained. Conservative analysis of what happens to the solidity constant ε' is not difficult.

Let the sending model $[X_-, X_+]$ be ε-regular, and the change in the inner and outer sets is bounded by some constant Δ. In other words, if the outer did not grow by more than Δ and the inner did not shrink by more than Δ, we have

$$k_\Delta(X'_-) \supseteq X_-, \qquad \text{and} \qquad k_\Delta(X_+) \supseteq X'_+ \tag{9}$$

These conditions together imply that $k_{\varepsilon'}(X'_-) \supseteq X'_+$, where $\varepsilon' = 2 * \Delta + \varepsilon$. Similarly, if the inner did not grow by more than Δ and the outer did not shrink by more than Δ, then

$$i_\Delta(X_-) \subseteq X'_-, \qquad \text{and} \qquad i_\Delta(X'_+) \subseteq X_+, \tag{10}$$

and therefore $i_{\varepsilon'}(X'_+) \subseteq X'_-$ also with $\varepsilon' = 2 * \Delta + \varepsilon$. Intuitively, the received model satisfying Equations 9 and 10 is a *possibly* larger (might be tighter) set interval approximating the original one. It should be clear that if $i_\Delta(X_-)$ is non-empty, then inner X'_- is still non-empty. Also, if $k_\Delta(X_+)$ is bounded then outer X'_+ is still bounded. Therefore, we conclude that if $[X_-, X_+]$ is an ε-solid and the changes in inner and outer sets are bounded by Δ, then $[X'_-, X'_+]$

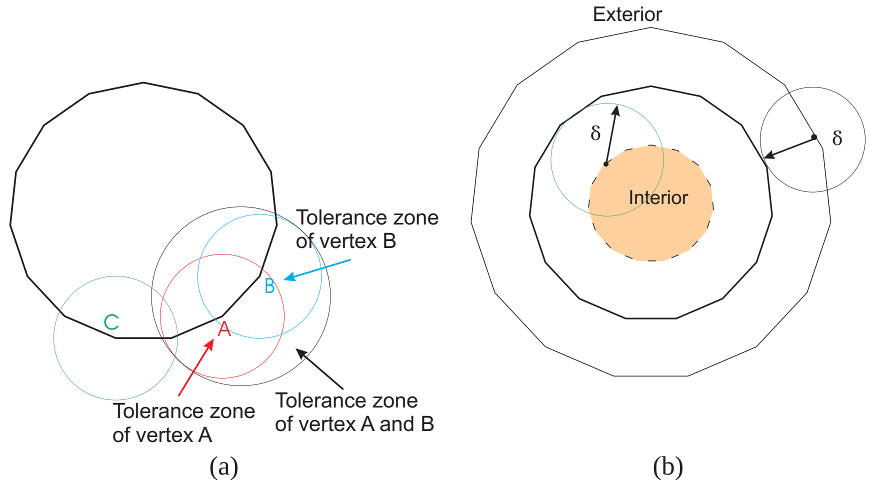


Figure 12: Solid validation procedure based on boundary representation is not sufficient under indicated tolerances: (a) vertex merging algorithm indicates that the representation is a single point, and hence is invalid; (b) non-empty i_δ and k_δ indicate a valid boundary representation of a solid.

is guaranteed to be an ε'-solid where $\varepsilon' = 2 * \Delta + \varepsilon$. The above is a conservative worst case analysis, since a direct comparison of X'_- and X'_+ may in fact give a smaller regularity constant ε'.

Accuracy is fixed $\lambda = \lambda'$, precision is changed $\delta \neq \delta'$. The geometric model is transferred exactly, but the two systems evaluate the model with different precisions (see Example 3 in Section 2.1). In current practice, the change of precision δ often triggers "healing" which is neither necessary nor sufficient (see section 4.2). In contrast, the following analysis shows that ε-solidity is usually guaranteed in this case.

If the precision of the receiving system is higher than that of the sending system ($\delta' < \delta$, with $\delta' \geq \lambda'$), a tighter set interval $[X'_-, X'_+]$ is induced for the model in the receiving system. Based on Theorem 3.13 we *know* that the model must be a valid ε'-solid with $\varepsilon' \leq \varepsilon$ being smaller than the original value of ε in the sending system. Specifically, let the model $[X_-, X_+]$ in the sending system be ε-regular, and let translation satisfy

$$i_\Delta(X'_-) \supseteq X_-, \qquad \text{and} \qquad k_\Delta(X'_+) \subseteq X_+ \tag{11}$$

Informally, we assume that a tighter precision δ' grows the inner by at least Δ and shrinks the outer by at least Δ. In this case, it is straightforward to show that with $\varepsilon' = \varepsilon - 2 * \Delta$, the conditions $k_{\varepsilon'}(X_-) \supseteq X_+$ and $i_{\varepsilon'}(X_+) \subseteq X_-$ are satisfied. This guarantees that $[X'_-, X'_+]$ is ε'-regular. Furthermore, if the original model $[X_-, X_+]$ is an ε-solid, then the received model $[X'_-, X'_+]$ is also guaranteed to satisfy the conditions of non-empty inner and bounded outer. Thus, if the original model was an ε-solid, the received model is guaranteed to be an ε'-solid. (By definition, the received model is also an ε-solid, but we are usually interested in the tightest possible bound.)

When the precision of the receiving system is lowered ($\delta' > \delta$), a larger set interval $[X'_-, X'_+]$ is induced for the model in the receiving system. Based on Theorem 3.13 we *know* that the model is a valid ε'-solid , but for some possibly larger and computable value ε', if X_- is still non-empty. The bound of $\varepsilon' = 2 * \Delta + \varepsilon$ can be estimated following a procedure analogous to the previous case of changing accuracy λ. However, in contrast to the previous worst case analysis, the set interval $[X'_-, X'_+]$ is guaranteed to grow whenever δ' increases.

Accuracy is changed $\lambda \neq \lambda'$, precision is changed $\delta \neq \delta'$. This is the most general and most difficult case of geometric data translation. The received model is affected by a composition of data errors and changes in the algorithm precision. The solution in this case will be a combination of techniques used in the special cases identified above.

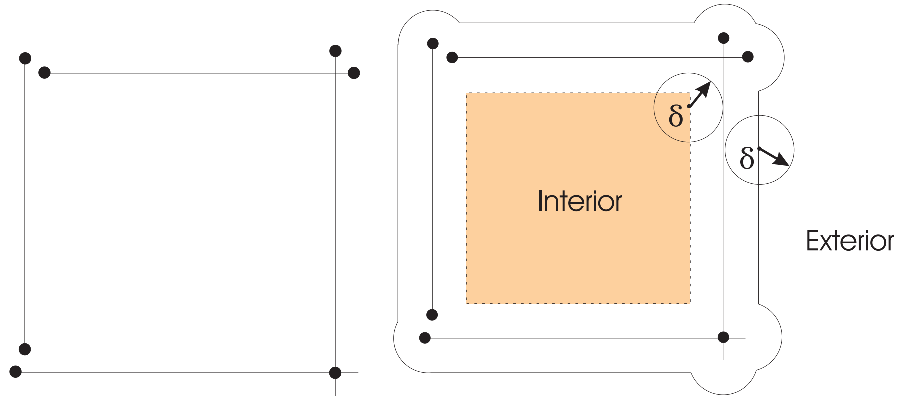


Figure 13: Repair of boundary representation is not necessary to assure validity under tolerances: (a) common errors in boundary representation indicate invalid object; (b) a suitable choice of PMC_δ procedure can classify points of the interior and of the exterior in the presence of errors.

5 Conclusions

5.1 Summary and Implications

This research started several years ago with a relatively modest goal of formulating and solving several specific problems in geometric data translation. Since the classical theory of solid modeling assumes exact geometric data and algorithms, it became clear that it must be substantially generalized before it could be applied to such problems. This paper proposes such a generalization, based on the observation that topological properties of sets may be represented and computed only within some finite precision. The formulated notions of ε-regularity, ε-solidity, and PMC_δ capture more realistically the practices and the recognized limitations of geometric and solid modeling. Importantly, the new theory subsumes the classical solid modeling formulations as a special (but not very realistic) case of $\varepsilon = 0$. In this sense, the proposed formulation of the problem certainly does not preclude exact representations and computations whenever such are feasible and practical. Nor is the theory limited by any particular choice of exact constants δ or ε. For example, it may be convenient to visualize and identify the δ-topological operations with the classical offsetting operations in solid modeling[42]; but in practice, PMC_δ is rarely implemented for any fixed value of δ; the discussion and observations in section 4 still apply following Theorem 3.11, as long as the interval $[X_-, X_+]$ is computed conservatively with $i_\delta \subseteq X_-$ and $k_\delta \supseteq X_+$.

We demonstrated that the proposed formulation allows systematic classification and investigation of problems in geometric data translation. In particular, the theory suggests that many current methods for validity checking of boundary representations are neither necessary nor sufficient for maintaining ε-solidity in the presence of numerical inaccuracies, whereas geometric healing procedures may be avoided in many common situations.

A number of practical steps may be taken immediately in order to alleviate the problems in geometric data translation. For example, a widely practiced technique of decreasing precision by increasing δ of the receiving system in order to make a model valid is a simple implementation of the requirement that $\delta \geq \lambda$; of course, this may also increase ε, producing a substantially different solid. When the precision δ of the receiving system is known a priori (and this is usually the case), a known valid ε-solid model may be simulated in the *sending* system for validity with different precisions δ.

In a longer term, the proposed revision of the classical solid modeling paradigm recognizes explicitly that accuracy λ of representation R and precision δ of PMC algorithms cannot and should not be considered separately from each other. Our observations and formulation point the way to improved redesign of both data structures and algorithms for solid modeling that explicitly recognize the distinct semantic roles of three physical constants: accuracy λ, precision δ, and solidity tolerance ε. The need for such a redesign has been apparent for some time, as witnessed by numerous efforts to deal with accuracy problems in STEP models and translations. Because our formulation does not require existence of exact valid objects (such as r-sets or manifolds), a number of robustness or validation problems may be easier than they appear. For example, Figure 10(f) shows that a PMC$_\varepsilon$ does not need to resolve dangling boundaries or other imperfections that are within ε of the interior i_δ of the solid. Thus, properly redefined ε-regularized set operations can be used to keep track and control the errors near solid's boundary.

5.2 Future Directions

Our observations suggest that modifying a geometric representation in order to find some imaginary 'correct' solid may not be a good idea in most circumstances. If anything needs to be 'healed' on the receiving end, it probably should be the topology and not geometry, in recognition that different choices of constants δ and ε may lead to substantially distinct topological interpretations. This statement may be extrapolated to more general tasks of simplification and small feature removal, such as those required in finite element meshing. A possible approach to such tasks is to induce ε-solids for larger values of ε starting with the original geometric data. Our formulation allows estimating the Hausdorff distance between original and translated ε-solid, but this gives no other guarantees on consistency of the result. As we explained in section 2.3, validity of the translation and ε-solidity in particular, provide a starting point for dealing with issues of translation consistency that are necessarily application specific. We have not considered these issues in this paper.

A significant feature of the proposed formal framework is that it is mostly representation free, in a sense that it does not assume any particular representation, approximation, or discretization of the represented pointsets. Application of the proposed formulation in specific representational problems should shed useful interpretations and establish relationship between seemingly unrelated techniques. For example, Delaunay-based solid reconstruction methods[4] and voxel-based approximation techniques[44, 36] appear to be directly related to problems of ε-solidity.

The concepts of ε-topological operations and ε-regularity may be also useful in formalizing semantics of geometric dimensioning and tolerancing (GD&T). Previous formulations proposed that a toleranced mechanical part is a class of regular sets[41] that contains an exact nominal set, as well as special perfect bounding (least and maximal) elements. Such a class of sets itself must satisfy additional metric and regularity conditions[14]. Generating and testing sets for membership in a tolerance class remains an open problem, particularly because the proposed definitions do not take into account limited resolution of inspection. In contrast, our approach using ε-topological operations may be more effective, because it would not require (but would allow) knowledge of any perfect regular elements while explicitly taking into account limited precision of representation and inspection.

A number of theoretical issues remain open. Accepting ε-topological operations and ε-solidity as foundations for solid modeling requires non-trivial revision of the usual concepts and operations. These include ε-continuity, ε-homeomorphism, and ε-regularized set operations. For example, definitions of ε-regularized set operations may appear straightforward, based on the definitions in this paper. However, such operations may not posses desired algebraic properties, such as obeying the distributive law. Many of the difficulties stem from recognition that an ε-solid is really a set interval. It is reasonable to expect that interval analysis [33] must form a basis for any algebraic system of ε-solids, but substantial enhancements are likely to be needed before such a system is useful for solid modeling. Such an algebraic system should have applications beyond solid modeling, e.g. in the area of robust geometric computations. Connections with computable analysis[55], constructive analysis[11], and domain theory[2] are also of considerable interest.

Acknowledgments

This research is supported in part by the National Science Foundation grants DMI-0115133, and CCR-0112758, DMI-0323514, and the National Institute of Standards (NIST) grant 60NANB2D0126, and UG PLM Solutions. The authors are grateful to Neil Stewart and Ralph Martin for carefully reading the earlier version of the paper and suggesting a number of improvements. Responsibility for errors and omissions lies solely with the authors.

References

[1] I. S. O. 10303. *Industrial Automation Systems and Integration - Product Data Representation and Exchange*. 1994.

[2] S. Abramsky and A. Jung. Domain theory. In S. Abramsky, D. Gabbay, and T. Maibaum, editors, *Handbook of Logic in Computer Science*, volume 3, pages 1–168. Clarendon Press, 1994.

[3] ACIS. *ACIS Documentation, Version 7*. Spatial Technologies, Boulder Heights, CO, 1998.

[4] N. Amenta and M. Bern. Surface reconstruction by voronoi filtering. *Discrete and Computational Geometry*, 22:481–504, 1999.

[5] F. Arbab. Set models and boolean operations for solids and assemblies. Technical report, University of Southern California, 1985.

[6] C. Armstrong, A. Bowyer, S. Cameron, J. Corney, G. Jared, R. Martin, A. Middleditch, M. Sabin, and J. Salmon. *Djinn: A geometric interface for solid modeling*. Information Geometers, 2000.

[7] G. Barequet, C. Duncan, and S. Kumar. RSVP: A geometric toolkit for controlled repair of solid models. *IEEE Transactions on Visualization and Computer Graphics*, 4(2):162–177, 1998.

[8] G. Barequet and M. Sharir. Filling gaps in the boundary of a polyhedron. *Computer Aided Geometric Design*, 12(2):207–229, 1995.

[9] R. Bidaara and W. Bronsvoort. Semantic feature modeling. *Computer-Aided Design*, 32(3):201–225, 2000.

[10] R. Bidaara, K. J. de Kraker, and W. Bronsvoort. Representation and management of feature information in a cellular model. *Computer-Aided Design*, 30(4):301–313, 1998.

[11] E. Bishop and D. Bridges. *Constructive Analysis*. Springer-Verlag, Berlin, 1985.

[12] J. H. Bohn. Removing zero-volume parts from CAD models for layered manufacturing. *IEEE Computer Graphics and Applications*, 15(6):27–34, 1995.

[13] J. H. Bohn and M. J. Wozny. A topology-based approach for shell-closure. In P. R. Wilson, M. J. Wozny, and M. J. Pratt, editors, *Geometric Modeling for Product Realization*, pages 297–319. North-Holland, Amsterdam, 1993.

[14] M. Boyer and N. F. Stewart. Modeling spaces for toleranced objects. *International Journal of Robotics Research*, 10(5):570–582, 1991.

[15] S. Brunnermeier and S. Martin. Interoperability cost analysis of the U.S. automotive supply chain. Technical report, Research Triangle Institute, Research Triangle Park, NC, Mar. 1999.

[16] G. Butlin and C. Stops. CAD data repair. In *5th International Meshing Roundtable*, pages 35–41, Sandia National Laboratories, 1996.

[17] E. Cech. *Topological Spaces*. Wiley, London, 1966.

[18] X. Chen and C. M. Hoffmann. Towards feature attachment. *Computer-Aided Design*, 27:675–702, 1995.

[19] A. L. Clark and S. M. Staley. STEP AP203 data exchange study. In *Proceedings of the 3rd ACM Symposium on Solid Modeling and Applications*, pages 213–224, Salt Lake City, Utah, May 1995.

[20] A. Edalat and A. Lieutier. Foundation of a computable solid modeling. In *Proceedings of the 5th ACM Symposium on Solid modeling and Applications*, pages 278–284, Ann Arbor, Michigan, 1999.

[21] A. Edalat and A. Lieutier. Foundation of a computable solid modelling. *Theoretical Computer Science*, 284(2):319–345, 2002.

[22] S. Fang, B. Bruderlin, and X. Zhu. Robustness in solid modeling: A tolerance-based intuitionistic approach. *Computer-Aided Design*, 25(9):567–576, 1993.

[23] S. Fortune. Polyhedral modeling with exact arithmetic. In *Proceedings of the 3rd ACM Symposium on Solid Modeling and Applications*, Salt Lake City, Utah, May 1995.

[24] S. Frechette. Interoperabilitiy requirements for CAD data transfer in the AutoSTEP project. Technical report, NISTIR 5844, National Institute of Standards and Technology, Gaithersburg, MD, 1996.

[25] L. Guibas, D. Salesin, and J. Stolfi. Epsilon geometry: Building robust algorithms from imprecise computations. In *Proceedings of the fifth ACM Symposium on Computational Geometry*, pages 208–217, Saarbruchen, West Germany, 1989.

[26] C. M. Hoffmann. *Geometric and Solid Modeling: An Introduction*. Morgan Kaufmann Publishers, San Mateo, CA, 1989.

[27] C. M. Hoffmann and R. Juan. Erep, an editable, high-level representation for geometric design and analysis. In *Geometric and Product Modeling*. North-Holland, USA, 1993.

[28] D. J. Jackson. Boundary representation modelling with local tolerancing. In *Proceedings of the 3rd ACM Symposium on Solid Modeling and Applications*, pages 247–253, Salt Lake City, Utah, May 1995.

[29] J. Keyser, S. Krishnan, and D. Manocha. Efficient and accurate b-rep generation of low degree sculptured solids using exact arithmetic. In *Proceedings of the 4th ACM Symposium on Solid Modeling and Applications*, pages 42–55, Atlanta, Georgia, May 1997.

[30] F.-L. Krause, C. Stiel, and J. Lddemann. Processing of cad-datałconversion, verification and repair. In *Proceedings of the 4th ACM Symposium on Solid modeling and applications*, pages 248–254, Atlanta, Georgia, 1997.

[31] K. Kuratowski and A. Mostowski. *Set Theory*. North-Holland Publishing Co., Amesterdam, 1976.

[32] B. Mendelson. *Introduction to Topology*. Dover Publications, New York, 1990.

[33] R. E. Moore. *Methods and Applications of Interval Analysis, SIAM Studies in Applied Mathematics*. SIAM, Philadelphia, PA, 1979.

[34] T. M. Murali and T. A. Funkhouser. Consistent solid and boundary representations from arbitrary polygonal data. In *Proceedings of the 1997 Symposium on Interactive 3D Graphics*, pages 155–162, Providence, RI, Apr. 1997.

[35] Parasolid. *Parasolid Documentation*, *Version 15*. EDS, Maryland Heights, MO, 2003.

[36] T. Peters, J. Bisceglio, D. Ferguson, C. Hoffmann, T. Maekawa, N. Patrikalakis, T. Sakkalis, and N. Stewart. Computational topology for regular closed sets (within the i-tango project). *Topology Atlas Invited Contributions*, 9(1):1–12, 2004.

[37] T. J. Peters, N. F. Stewart, D. R. Ferguson, and P. S. Fussell. Algorithmic tolerances and semantics in data exchange. In *Proceedings of the thirteenth ACM symposium on Computational geometry*, pages 403–405, Nice, France, June 1997.

[38] S. Raghothama and V.Shapiro. Boundary representation deformation in parametric solid modeling. *ACM Transactions on Graphics*, 17(4):259–286, Oct. 1998.

[39] A. Rappoport. The generic geometric complex (GGC): a modeling scheme for families of decomposed pointsets. In *Proceedings of the 4th ACM Symposium on Solid Modeling and Applications*, pages 19–30, Atlanta, Georgia, May 1997.

[40] A. A. G. Requicha. Representations for rigid solids: Theory, methods and systems. *ACM Computing Surveys*, 12(4):437–464, Dec. 1980.

[41] A. A. G. Requicha. Towards a theory of geometric tolerancing. *International Journal of Robotics Research*, 2(4):45–60, 1983.

[42] J. R. Rossignac and A. A. G. Requicha. Offsetting operations in solid modelling. *Computer Aided Geometric Design*, 3(2):129–148, 1986.

[43] T. Sakkalis, G. Shen, and N. M. Patrikalakis. Representational validity of boundary representation models. *Computer Aided Design*, 32(12):719–726, Oct. 2000.

[44] T. Sakkalis, G. Shen, and N. M. Patrikalakis. Topological and geometric properties of interval solid models. *Graphical Models*, 63(3):163–175, 2001.

[45] M. Segal. Using tolerances to guarantee valid polyhedral modeling results. In *Computer Graphics (Proceedings of ACM SIGGRAPH 90)*, pages 105–114, Aug. 1990.

[46] V. Shapiro. Maintenance of geometric representations through space decompositions. *International Journal of Computational Geometry and Applications*, 7(4):383–418, 1997.

[47] V. Shapiro. Solid modeling. In G. Farin, J. Hoschek, and M.-S. Kim, editors, *Handbook of Computer Aided Geometric Design*, pages 473–518. Elsevier Science Publishers, 2002.

[48] V. Shapiro and D. L. Vossler. What is a parametric family of solids? In *Proceedings of the 3rd ACM Symposium on Solid Modeling and Applications*, Salt Lake City, Utah, May 1995.

[49] G. Shen, T. Sakkalis, and N. M. Patrikalakis. Analysis of boundary representation model rectification. In *Proceedings of the 6th ACM Symposium on Solid Modeling and Applications*, pages 149–158, Ann Arbor, Michigan, 2001.

[50] X. Sheng and I. R. Meier. Generating topological structures for surface models. *IEEE Computer Graphics and Applications*, 15(6):35–41, 1995.

[51] S. Spitz and A. Rappoport. Integrated feature-based and geometric CAD data exchange. In *Proceedings of ACM Symposium on Solid modeling and Applications (2004)*, Genova, Italy, 2004.

[52] S.Raghothama and V.Shapiro. Topological framework for part families. *ASME Journal of Computing and Information Sciences*, 32(8-9), Mar. 2003.

[53] R. B. Tilove. Set membership classification: A unified approach to geometric intersection problems. *IEEE Transactions on Computer*, C-29(10), Oct. 1980.

[54] K. Weihe and T. Willhalm. Reconstructing the topology of a CAD model - a discrete approach. *Algorithmica*, 26(1):126–147, 2000.

[55] K. Weihrauch. *Computable analysis: an introduction*. Springer-Verlag, Berlin, 2000.

[56] C. Yap. Towards exact geometric computation. *Computational Geometry: Theory and Applications*, 7(1-2):3–23, 1997.