Geometric Interoperability With Epsilon Solidity

Jianchang Qi
e-mail: qi@students.wisc.edu

Vadim Shapiro
e-mail: vshapiro@engr.wisc.edu

Spatial Automation Laboratory,
Department of Mechanical Engineering,
University of Wisconsin–Madison,
1513 University Avenue,
Madison, WI 53706

Geometric data interoperability is critical in industrial applications where geometric data are transferred (translated) among multiple modeling systems for data sharing and reuse. A big obstacle in data translation lies in that geometric data are usually imprecise and geometric algorithm precisions vary from system to system. In the absence of common formal principles, both industry and academia embraced ad hoc solutions, costing billions of dollars in lost time and productivity. This paper explains how the problem of interoperability, and data translation in particular, may be formulated and studied in terms of a recently developed theory of ε-solidity. Furthermore, a systematic classification of problems in data translation shows that in most cases ε-solids can be maintained without expensive and arbitrary geometric repairs. [DOI: 10.1115/1.2218367]

Keywords: interoperability, geometric data translation, validity and consistency, solid modeling, ε-solidity

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. The 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 [1]. A typical geometric data translation problem between two systems is illustrated in Fig. 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 Fig. 1 subsumes many other types of translation problems. For example, the classical problems of boundary evaluation, boundary representation (BRep) to constructive solid geometry (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 the mismatch between the accuracy of the 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 a result of such 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 [2], because such representations are largely symbolic structures with 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 [3–12], 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.

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 20 years ago [13]. 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 nontrivial solid has a nonempty 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 [13].

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

[1]See Section 3.2 for additional discussion and examples.

Contributed by the Computer Aided Product Development (CAPD) Committee of ASME for publication in the JOURNAL OF COMPUTING AND INFORMATION SCIENCE IN ENGINEERING. Manuscript received August 9, 2004; final manuscript received August 1, 2005. Assoc. Editor: N. Patrikalakis.

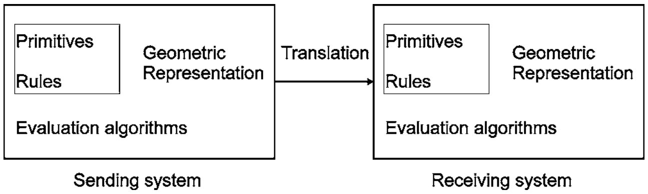

Fig. 1 A generic geometric data translation diagram

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 Sec. 2.

The fundamental difficulty with all proposed solutions is that they are not based on a suitable formal model. In particular, we will demonstrate in Sec. 3 that the classical notions of solid representation validity and consistency (based on the assumption of exactness) are too simplistic for data translation problem. As implied by the diagram in Fig. 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 describe a recently developed theory of ε-solidity that is designed to deal with such approximation and precision errors, and apply it to several specific common instances of the data translation problem.

1.2 Approach and Outline. After briefly summarizing in Sec. 2 previous efforts on geometric data translation, in Sec. 3 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 approximations of primitives and/or changes in the precision of evaluation algorithms.

In Sec. 4, we first summarize the theory of ε-solidity [14,15]. This theory relies on finite size neighborhoods, ε-topological operations, and ε-regularity to formulate the notion of ε-solidity in order to deal with issues of data accuracy and algorithm precision in geometric modeling. We show that, under the proposed model, a systematic classification of common data translation problems suggests that in many cases validity may be maintained without healing. Open issues and promising extensions of the proposed approach are considered in Sec. 5.

2 Related Work

A vast amount of literature exists on the 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 case 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 [16–18]. 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 (we take the data translation among multiple systems as a sequence of data translations between respective two systems).

Several other proposals attempted to deal with imprecision of data and/or algorithms. Notably, Guibas, et al. [19] 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 modeling and translation problems because they assume that geometric data are exact and 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 a 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 *approximate interval solid* [22] arises naturally from numerical considerations and was used to repair defective geometric models [23] and to develop robust algorithms for intersection of surfaces [24].

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 [25–27], 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 (combinatorial) structure 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 [25,26] that some choices of tolerance values may lead the proposed algorithm to the erroneous answers. Commercial software systems such as ACIS [28] and Parasolid [29] 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 were 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

[2]In othor 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 to the newer paper [21] for discussion.

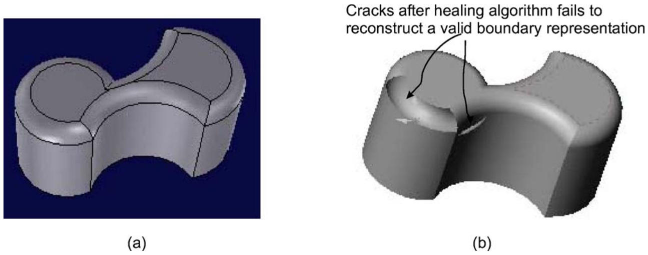

Fig. 2 Even minor changes in geometric primitives during translation may invalidate the model: (*a*) original model and (*b*) a failed attempt to repair the translated model

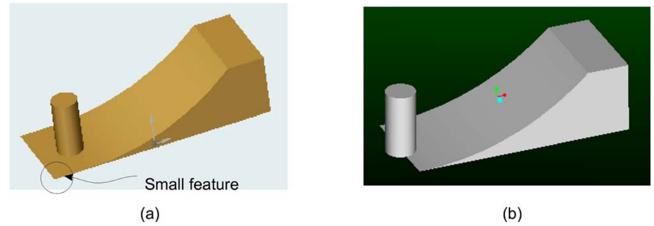

Fig. 4 Evaluating the same geometric data with different precisions results in inconsistent solids: (*a*) the original model with small feature in the sending system and (*b*) the received model with small feature removed after healing

have been proposed for linear polyhedral models [30–34] as well as for more general solids [23,35,36]. Perturbation techniques [23,31,35] 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 [30–32] and/or allow substantial changes in topological structure for the sake of repair. A more successful example 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 reevaluating the boundary of the union of all solid cells that are judged (heuristically) to be 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 [23,30]. 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 are translated into yet another receiving system, they may become invalid and in need of healing again; if the repaired data are translated back into the original sending system, they 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 require the existence of an exact *r*-set**. Under this model, we showed in [14] that some popular tolerant modeling algorithms are not sufficient, and that typical repair algorithms are not necessary for maintaining the solidity of boundary representations.

3 Formulation of Data Translation

3.1 Motivating Examples. Many references [1,37–41] have illustrated various data translation problems. We will not attempt to add to the long list of well-known difficulties, but rather consider a 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 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 Fig. 2(*a*) is created in SolidWorks and saved in the STEP (STandard for the Exchange of Product model data) neutral data exchange file format [42]. 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 Fig. 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 Fig. 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]

Example 2. The second example (Fig. 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 Fig. 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.

Example 3. The third example shows that differences in precision of evaluating algorithms are also key ingredients of the translation difficulties, even when the changes in geometric representations are negligible. The solid model in Fig. 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 evaluated in Pro-Engineer. It is certainly a valid solid, but with a drastically different shape that

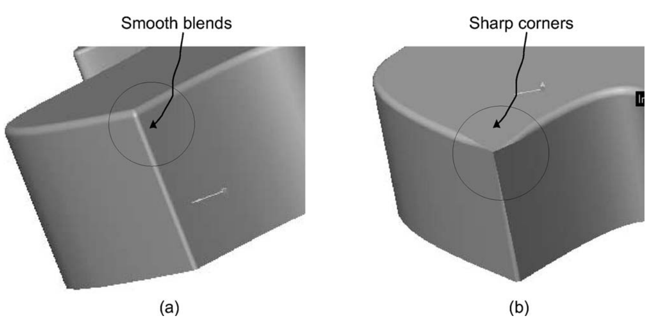

Fig. 3 Healing algorithms may drastically change important geometric properties: (*a*) the original model with smooth blends and (*b*) an automatically repaired model with sharp corners

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

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 m, matching the minimum thickness of the model in Fig. 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 Fig. 4(*b*).

3.2 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 Fig. 1, is born out by the examples, and includes classical solid representation conversion problems. For example, when boundary representation is evaluated from 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 boundary representation. 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]

3.3 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, 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 [13] 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 Fig. 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 specific comparison criteria.* Once again, this definition is axiomatic; it recognizes explicitly that the notion of consistency depends on the 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.

4 Data Translation With ε-Solids

In this section, we briefly summarize the theory of ε-solidity developed by the authors in [14] and demonstrate how it can be applied to the data translation problem.

4.1 Theory of ε-Solids. Classical solid modeling theory builds upon the notion of *r*-sets, defined as bounded, semi-analytic, and closed regular subsets of E^3 [13]. This definition of a solid is "ideal" in the sense that a solid is a set of points with dimensionally homogeneous interior and well-defined boundary, and a neighborhood of every boundary point contains points in the set interior as well as points in its exterior. However, in real applications, we are unlikely to compute this exact single set X even

[5] It can be argued that all geometric algorithms sooner or later reduce to a finite number of point membership tests [43].

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

when using an exact representation such as CSG, because of the use of finite precision algorithms, and in most cases the representations themselves are inaccurate. In [14], we argue that a more appropriate mathematical model of solid is ε-solid defined as follows:

DEFINITION 4.1. An *ε-solid* is an ε-regular set interval $[X_-, X_+]$ with nonempty X_- and bounded X_+.

This definition requires new concepts of *ε-regularity* and *set interval*, which are explained below. Intuitively, we replace the notion of a single set by a family of sets that are in some precise sense close to each other. This is justified by the pragmatic view that whenever the distance between sets is smaller than the size of errors, the sets should be viewed as one and the same solid. Formally, a *set interval* $[X_-, X_+]$ is the class of sets $\{X\}$ such that $X_- \subseteq X \subseteq X_+$, where X_- is an open set and X_+ is a closed set. X_- is called *inner*, and X_+ is called *outer*. A *subinterval* $[Y_-, Y_+]$ of $[X_-, X_+]$ is the class of sets $\{Y\}$ such that $Y_- \subseteq Y \subseteq Y_+$, with the inner $Y_- \supseteq X_-$ and the outer $Y_+ \subseteq X_+$.

The notion of ε-regularity of a set interval is a generalization of the classical notions of open and closed regular sets. Traditionally, solidity is formulated in terms of the 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. In [14], we proposed 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. The definitions are straightforward generalizations of the classical operations.[7]

We then define an ε-regular interval similarly to regular sets, but using ε-topological operations and a pair of set inequalities in place of the usual equalities.

DEFINITION 4.2. A set interval $[X_-, X_+]$ is *ε-regular* if, for a given non-negative real number ε,

$$i_\varepsilon(X_+) \subseteq X_- \subseteq X_+ \subseteq k_\varepsilon(X_-). \tag{1}$$

The above definition can have two special cases when $\varepsilon=0$: a closed regular set X has a bounding interval $[i_0(X), X]$ and satisfies $i_0(X_+)=X_- \subseteq X_+ = k_0(X_-)$; an open regular set X has a bounding interval $[X, k_0(X)]$ and satisfies $i_0(X_+)=X_- \subseteq X_+ = k_0(X_-)$. Intuitively, the definition of ε-regular interval requires that the errors of size less than ε are *tolerated* because they are hidden within the interval $[i_\varepsilon(X_+), k_\varepsilon(X_-)]$; the boundary of the solid becomes "thickened" to correspond to boundaries of all sets within the interval. In [15] we have shown that ε-regularity implies that the Hausdorff distances between X_- and X_+, their boundaries, and their complements are all bounded by ε.

The above definitions of ε-regularity and ε-solidity generalize and subsume the classical definitions of regularity (both open and closed) and solidity, which correspond to the special nominal case of $\varepsilon=0$. The notion of ε-solidity formalizes the concept of validity for interchangeable geometric models, providing both the notion and the *measure* of geometric variability ε that can be used simultaneously at different levels of resolution. For example, it is not difficult to show that:

THEOREM (SUBINTERVAL SOLIDITY). *Any subinterval $[Y_-, Y_+]$ of ε-solid $[X_-, X_+]$ is ε-solid.*

This result is of paramount practical significance because it allows to verify the solidity of interval $[Y_-, Y_+]$, even when the interval itself is not computable, by testing a larger containing interval $[X_-, X_+]$ that is computable. Additional details and properties of ε-regular sets and ε-solids are studied in [14].

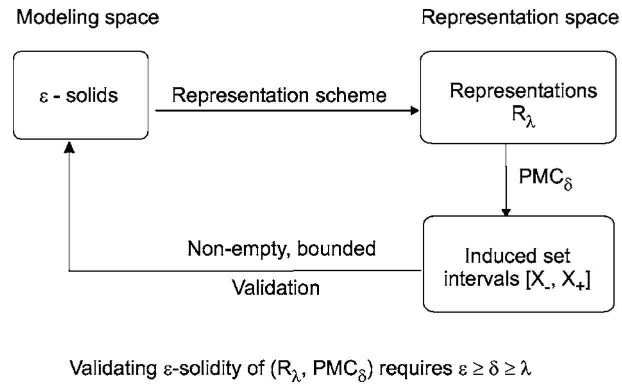

Fig. 5 Modeling and representation spaces for ε-solids with limited data accuracy λ and algorithm precision δ

The theory of ε-solidity allows a precise and systematic formulation of problems involving data accuracy λ and algorithm precision δ. Consider one of the most basic tasks of assigning semantics to a given geometric representation (CSG, boundary representation, etc.). A fundamental property of any unambiguous representation is its ability to support PMC queries to determine if a given point is inside, outside, of boundary of the represented solid. Assuming that the precision of the PMC algorithm is δ, the PMC_δ test with respect to an exactly represented X is defined in terms of the ε-topological operations as:

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

In other words, the semantics of the representation is defined by the interval $[i_\delta(X), k_\delta(X)]$. This induced interval can only be ε-regular with $\varepsilon \geqslant \delta$. If a geometric representation R_λ of X has errors of size λ, the corresponding PMC procedure has to have its accuracy $\delta \geqslant \lambda$ to maintain the assumed semantics. For example, if a representation is supposed to be connected within some resolution λ, then applying PMC with smaller δ may invalidate the connectedness semantics. Figure 5 shows the general paradigm of ε-solid modeling, where the modeling space consists of ε-solids, and the representation space consists of pairs of (R_λ, PMC_δ). Under this paradigm, we say that a representation (R_λ, PMC_δ) is *valid* if application of PMC_δ on R_λ induces an ε-solid.

4.2 Maintaining ε-Solids in Data Translation. The proposed validity formulation of geometric data translation problem relies on the geometric model validity in both the sending system and the receiving system, as described in Sec. 3. Thus, we will *assume* that a geometric model is valid in the sending system, and ask *Under what conditions is the translated model valid in the receiving system*? The term "valid" can now be interpreted to mean that "a geometric model with accuracy λ is an ε-solid with respect to a suitable PMC_δ algorithm with precision δ." Let λ, δ, ε be data accuracy, algorithm precision, and solidity measure in the sending system respectively, 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:

4.2.1 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 the same precision of evaluation algorithms,

[7]For example, a point x is said to be in the *ε-closure* of set X, denoted $k_\varepsilon(X)$, if for every $r > \varepsilon$, an open ball $B(x, r)$ of radius r centered at x intersects the set X. The operations of *ε-interior i_ε, ε-exterior e_ε and ε-boundary ∂_ε* are defined similarly.

and there are no approximations in translation. Then, if the model X is an ε-solid in the sending system, it will remain to be an ε-solid in the receiving system.

4.2.2 Accuracy is Changed $\lambda \neq \lambda'$, Precision is Fixed $\delta = \delta'$. Both systems evaluate the models with the same algorithm precision, but the 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 Sec. 3.1), or when a 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 $\text{PMC}_{\delta'}$ procedure may also need to be increased to maintain $\delta' \geqslant \lambda'$ 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 the condition $\delta' \geqslant \lambda'$ is satisfied, the receiving system induces a different set interval $[X'_-, X'_+]$ from the original $[X_-, X_+]$ in the sending system. If the inner X'_- becomes empty or the outer X'_+ becomes unbounded, then the received model cannot be a valid solid. However, in most practical situations, the received model approximates closely to the sending model, and the conditions of nonempty inner and bounded outer are usually maintained. A conservative analysis of what happens to the solidity constant ε' is not difficult, as shown below.

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 inner X_- does not grow by more than Δ from X'_-, and the outer X'_+ does not grow by more than Δ from X_+, we have

$$k_\Delta(X'_-) \supseteq X_- \quad \text{and} \quad k_\Delta(X_+) \supseteq X'_+. \tag{2}$$

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

$$i_\Delta(X'_+) \subseteq X_+ \quad \text{and} \quad i_\Delta(X_-) \subseteq X'_-, \tag{3}$$

and therefore $i_{\varepsilon'}(X'_+) \subseteq X'_-$ also with $\varepsilon' = 2*\Delta+\varepsilon$.

Intuitively, the received model satisfying Eqs. (2) and (3) is a *possibly* larger (might be tighter) set interval approximating the original one. It should be clear that if $i_\Delta(X_-)$ is nonempty, then the inner X'_- is still nonempty. Also, if $k_\Delta(X_+)$ is bounded, then the outer X'_+ is still bounded. Therefore, we conclude that if $[X_-, X_+]$ is an ε-solid and the changes in the inner and outer sets are bounded by Δ, then $[X'_-, X'_+]$ is guaranteed to be an ε'-solid with $\varepsilon' = 2*\Delta+\varepsilon$. 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'.

4.2.3 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 Sec. 3.1). In current practice, the change of precision δ often triggers "healing," which is neither necessary nor sufficient [14]. 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 $\delta' \geqslant \lambda'$), a tighter set interval $[X'_-, X'_+]$ is induced for the model in the receiving system. Based on the subinterval solidity theorem, we *know* that the model must be a valid ε'-solid with $\varepsilon' \leqslant \varepsilon$ 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 the translation satisfy

$$i_\Delta(X'_-) \supseteq X_- \quad \text{and} \quad k_\Delta(X'_+) \subseteq X_+. \tag{4}$$

Then, in the receiving system 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 the conditions $k_{\varepsilon'}(X'_-) \supseteq X'_+$ and $i_{\varepsilon'}(X'_+) \subseteq X'_-$ are satisfied, with $\varepsilon' = \varepsilon - 2*\Delta$. 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 nonempty inner and bounded outer. Thus, if the original model is 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 the subinterval solidity theorem, we *know* that the received model is a valid ε'-solid if X'_- is still nonempty and X'_+ is still bounded, but for some possibly larger and computable value ε'. 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.

4.2.4 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.

5 Conclusions

5.1 Summary. We demonstrated that the proposed formulation allows systematic classification and investigation of problems in geometric data translation. In particular, the theory of ε-solidity 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 $\delta \geqslant \lambda$; 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.

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 Sec. 3.3, 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.

5.2 Significance in a Broader Context. It is generally accepted that modern mass production and most of the manufacturing technologies of the past century would not be possible without the concept of *interchangeable* parts [44]. The doctrine of interchangeability dictates that a mechanical part may be replaced by another "equivalent" component without affecting the overall function of the product.[8] Prior to the adoption of this principle, manufacturing was a custom art practiced mainly by skilled artisans on a small scale. Implementation of the principle of interchangeability required a definition of part equivalency; early practices relied on functional gauges that later were replaced by standardized principles of *tolerancing* (focused on specification and control of geometric variability) and *metrology* (inspection of manufactured parts through measurement and analysis of errors). Today, tolerancing and metrology are mature disciplines and active areas of research—driven partly by the emergence of new manufacturing technologies and partly by the eternal drive to improve our notion of "equivalent" parts and behaviors.

With the emergence of computer-aided design and manufacturing over the last 40 years, most engineering tasks today are performed virtually, by simulating them on computer representations in place of physical parts and processes. One could argue that *virtual engineering has become an enterprise for manufacturing virtual components themselves.* The object of manufacturing, in this case, is the computer model of a physical artifact, and the manufacturing processes are the above computer transformations involved throughout the life cycle of this model. It is our belief that tolerancing and metrology of interchangeable virtual components is as important to the future of virtual engineering, as interchangeability of mechanical components was critical for emergence of mass production and modern manufacturing practice.

As with all manufacturing processes, virtual engineering is inherently imprecise due to limited data accuracy and approximate geometric computations. However, virtual engineering differs substantially from other manufacturing technologies because (1) it is a highly customized production, where each component is somewhat different, and statistical methods may not apply; (2) virtual engineering involves highly repetitive manufacturing processes, because many components undergo similar or identical manufacturing transformations; (3) geometric accuracy and algorithm precision of the same component vary widely and concurrently between processes and systems; and (4) in contrast to real mechanical components, virtual components cannot be assumed to be valid objects.

The first observation may seem to suggest that interchangeability of virtual components is not an important issue. The second point implies that the critical bottleneck in manufacturing of virtual components is interoperability: the ability to rapidly transfer a given virtual component between a variety of manufacturing processes. But the thrid characteristic of virtual engineering explains why achieving interoperability is difficult: interoperability demands interchangeability of different *instances* of the same component in the presence of errors and approximations. The final issue of component validity is clearly a prerequisite for *any* solution to the problem of interchangeability. We hope that this paper offers the first step in that direction.

Acknowledgment

This research is supported in part by the National Science Foundation Grant Nos. DMI-0500380, DMI-0323514, DMI-0115133, and CCR-0112758, the National Institute of Standards and Technology (NIST) Grant No. 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.

[8]First articulation of this principle is often attributed to Whitney in the context of manufacturing of fire arms, but it apparently was also employed by others much earlier, for example in the manufacture of clocks at the beginning of the 1700s [45].

References

[1] Brunnermeier, S. B., and Martin, S. A., 1999, "Interoperability Cost Analysis of the U. S. Automotive Supply Chain," Tech. Rep., Research Triangle Institute, Research Triangle Park, NC, March.

[2] Spitz, S., and Rappoport, A., 2004, "Integrated Feature-Based and Geometric CAD Data Exchange," Proceedings of the 9th ACM Symposium on Solid Modeling and Applications, Genova, Italy, pp. 183–190.

[3] Shapiro, V., 1997, "Maintenance of Geometric Representations Through Space Decompositions," Int. J. Comput. Geom. Appl., **7**(4), pp. 383–418.

[4] Hoffmann, C. M., and Juan, R., 1993, "Erep: An Editable, High-Level Representation for Geometric Design and Analysis," in *Geometric Modeling for Product Realization*, P. R. Wilson, M. J. Wozny, and M. J. Pratt, eds., North-Holland, Amsterdam, pp. 129–164.

[5] Chen, X., and Hoffmann, C. M., 1995, "Towards Feature Attachment," Comput.-Aided Des., **27**(9), pp. 695–702.

[6] Shapiro, V., and Vossler, D. L., 1995, "What is a Parametric Family of Solids?" Proceedings of the 3rd ACM Symposium on Solid Modeling and Applications, Salt Lake City, Utah, pp. 43–54.

[7] Rappoport, A., 1997, "The Generic Geometric Complex (GGC): A Modeling Scheme for Families of Decomposed Pointsets," Proceedings of the 4th ACM Symposium on Solid Modeling and Applications, Atlanta, Georgia, pp. 19–30.

[8] Armstrong, C., Bowyer, A., Cameron, S., Corney, J., Jared, G., Martin, R., Middleditch, A., Sabin, M., and Salmon, J., 2000, *Djinn: A Geometric Interface for Solid Modeling*, Information Geometers, Winchester.

[9] Raghothama, S., and Shapiro, V., 1998, "Boundary Representation Deformation in Parametric Solid Modeling," ACM Trans. Graphics, **17**(4), pp. 259–286.

[10] Bidarra, R., de Kraker, K. J., and Bronsvoort, W. F., 1998, "Representation and Management of Feature Information in a Cellular Model," Comput.-Aided Des., **30**(4), pp. 301–313.

[11] Bidarra, R., and Bronsvoort, W. F., 2000, "Semantic Feature Modelling," Comput.-Aided Des., **32**(3), pp. 201–225.

[12] Raghothama, S., and Shapiro, V., 2002, "Topological Framework for Part Families," ASME J. Comput. Inf. Sci. Eng., **2**(4), pp. 246–255.

[13] Requicha, A. A. G., 1980, "Representations for Rigid Solids: Theory, Methods and Systems," ACM Comput. Surv., **12**(4), pp. 437–464.

[14] Qi, J., and Shapiro, V., 2004, "ε-Solidity in Geometric Data Translation," Tech. Rep. SAL 2004-2, Spatial Automation Laboratory, University of Wisconsin–Madison, June.

[15] Qi, J., and Shapiro, V., 2005, "Epsilon-Regular Sets and Intervals," IEEE International Conference on Shape Modeling and Applications, MIT, Cambridge, MA, pp. 308–317.

[16] Yap, C., 1997, "Towards Exact Geometric Computation," Comput. Geom., **7**(1–2), pp. 3–23.

[17] Fortune, S., 1995, "Polyhedral Modelling With Exact Arithmetic," Proceedings of the 3rd ACM Symposium on Solid Modeling and Applications, Salt Lake City, Utah, pp. 225–234.

[18] Keyser, J., Krishnan, S., and Manocha, D., 1997, "Efficient and Accurate B-rep Generation of Low Degree Sculptured Solids Using Exact Arithmetic," Proceedings of the 4th ACM Symposium on Solid Modeling and Applications, Atlanta, Georgia, pp. 42–55.

[19] Guibas, L., Salesin, D., and Stolfi, J., 1989, "Epsilon Geometry: Building Robust Algorithms From Imprecise Computations," Proceedings of the Fifth ACM Symposium on Computational Geometry, Saarbrüchen, West Germany, pp. 208–217.

[20] Edalat, A., and Lieutier, A., 1999, "Foundation of a Computable Solid Modeling," Proceedings of the 5th ACM Symposium on Solid Modeling and Applications, Ann Arbor, Michigan, pp. 278–284.

[21] Edalat, A., and Lieutier, A., 2002, "Foundation of a Computable Solid Modelling," Theor. Comput. Sci., **284**(2), pp. 319–345.

[22] Sakkalis, T., Shen, G., and Patrikalakis, N. M., 2001, "Topological and Geometric Properties of Interval Solid Models," Graphical Models, **63**(3), pp. 163–175.

[23] Shen, G., Sakkalis, T., and Patrikalakis, N. M., 2001, "Analysis of Boundary Representation Model Rectification," Proceedings of the 6th ACM Symposium on Solid Modeling and Applications, Ann Arbor, Michigan, pp. 149–158.

[24] Peters, T. J., Bisceglio, J., Ferguson, D. R., Hoffmann, C. M., Maekawa, T., Patrikalakis, N. M., Sakkalis, T., and Stewart, N. F., 2004, "Computational Topology for Regular Closed Sets (Within the I-TANGO Project)," Topol. Atlas Invited Contrib., **9**(1), pp. 1–12.

[25] Segal, M., 1990, "Using Tolerances to Guarantee Valid Polyhedral Modeling Results," Comput. Graph., **24**(4), pp. 105–114.

[26] Fang, S., Brüderlin, B., and Zhu, X., 1993, "Robustness in Solid Modeling: A Tolerance-Based Intuitionistic Approach," Comput.-Aided Des., **25**(9), pp. 567–576.

[27] Jackson, D. J., 1995, "Boundary Representation Modelling With Local Tolerancing," Proceedings of the 3rd ACM Symposium on Solid Modeling and Applications, Salt Lake City, Utah, pp. 247–253.

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

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

[30] Barequet, G., and Sharir, M., 1995, "Filling Gaps in the Boundary of a Polyhedron," Comput. Aided Geom. Des., **12**(2), pp. 207–229.

[31] Barequet, G., Duncan, C. A., and Kumar, S., 1998, "RSVP: A Geometric Toolkit for Controlled Repair of Solid Models," IEEE Trans. Vis. Comput. Graph., **4**(2), pp. 162–177.

[32] Bøhn, J. H., and Wozny, M. J., 1993, "A Topology-Based Approach for Shell-Closure," in *Geometric Modeling for Product Realization*, P. R. Wilson, M. J. Wozny, and M. J. Pratt, eds., North-Holland, Amsterdam, pp. 297–319.

[33] Bøhn, J. H., 1995, "Removing Zero-Volume Parts From CAD Models for Layered Manufacturing," IEEE Comput. Graphics Appl., **15**(6), pp. 27–34.

[34] Murali, T. M., and Funkhouser, T. A., 1997, "Consistent Solid and Boundary Representations From Arbitrary Polygonal Data," Proceedings of the 1997 Symposium on Interactive 3D Graphics, Providence, RI, pp. 155–162.

[35] Sheng, X., and Meier, I. R., 1995, "Generating Topological Structures for Surface Models," IEEE Comput. Graphics Appl., **15**(6), pp. 35–41.

[36] Weihe, K., and Willhalm, T., 2000, "Reconstructing the Topology of a CAD Model—a Discrete Approach," Algorithmica, **26**(1), pp. 126–147.

[37] Clark, A. L., and Staley, S. M., 1995, "STEP AP203 Data Exchange Study," Proceedings of the 3rd ACM Symposium on Solid Modeling and Applications, Salt Lake City, Utah, pp. 213–224.

[38] Frechette, S., 1996, "Interoperabilitiy Requirements for CAD Data Transfer in the AutoSTEP Project," Tech. Rep. NISTIR 5844, National Institute of Standards and Technology, Gaithersburg, MD.

[39] Butlin, G., and Stops, C., 1996, "CAD Data Repair," Proceedings of the 5th International Meshing Roundtable, Pittsburgh, PA, pp. 7–12.

[40] Krause, F.-L., Stiel, C., and Lüddemann, J., 1997, "Processing of CAD-Data—Conversion, Verification and Repair," Proceedings of the 4th ACM Symposium on Solid Modeling and Applications, Atlanta, Georgia, pp. 248–254.

[41] Peters, T. J., Stewart, N. F., Ferguson, D. R., and Fussell, P. S., 1997, "Algorithmic Tolerances and Semantics in Data Exchange," Proceedings of the 13th ACM Symposium on Computational Geometry, Nice, France, pp. 403–405.

[42] ISO 10303, 1994, *Industrial Automation Systems and Integration—Product Data Representation and Exchange*. ISO.

[43] Shapiro, V., 2002, "Solid Modeling," in *Handbook of Computer Aided Geometric Design*, G. Farin, J. Hoschek, and M.-S. Kim, eds., Elsevier Science, Amsterdam, pp. 473–518.

[44] Hounshell, D., 1984, *From the American System to Mass Production, 1800–1932*. The Johns Hopkins U. P.

[45] Woodbury, R., 1972, "The Legend of Eli Whitney and Interchangeable Parts," in *Technology and Culture: An Anthology*, M. Kransberg, and W. H. Davenport, eds., Schocken, New York, pp. 318–335.