Research in Engineering Design (1996) 2: 81–98

Research in
Engineering
Design

The Role of Topology in Engineering Design Research

D. W. Rosen and T. J. Peters

The George W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, USA and Department of Computer Science and Engineering, University of Connecticut, Storrs, USA

Abstract. *Aspects of the mathematical specialty of topology appear within several seemingly distinct areas of engineering design and engineering design theory. Indeed, the expression "topology of a design" is often used informally. In this article a primary intent is to demonstrate the diversity of applications of topology within engineering design. A complementary goal is to introduce the engineering design community to topology as a rich, formal, well-established mathematical discipline that may be of value for wider study. Upon reveiwing some of these topological applications, it appears that topology holds promise as a basis for formalizing engineering design theory. This article considers topology as a basis for unifying design abstractions. The potential benefit may be the realization of commonalities between design aspects previously considered separately, where each now has its own attendant specialized, expensive analyses.*

Keywords. Chain models; Design for manufacture; Design theory; Solid modeling; Topology; Tolerance Modeling

1. Introduction

Engineering design is in a pre-theory stage. Several communities of researchers are actively involved in the development of mathematical models to describe design and related product development activities. Ward (1990, 1992) has called for the development of a new mathematics for design, which includes reasoning about sets of candidate designs. Concurrent engineering requires the ability to communicate about sets of designs and sets of constraints among a diverse group of engineers, designers, manufacturing personnel, and other team members. Topology is applicable as it is rich in tools for formally expressing relations amongst sets. Tolerance issues are critical at the design–manufacturing interface. Emerging formal models of tolerance (Boyer and Stewart, 1991, 1992; Stewart, 1993; Andersson *et al.*, 1994) and of the design–manufacturing interface (Rosen and Peters, 1992; Peters *et al.*, 1994*c*) share fundamental topological concerns. The phrase "the topology of a design" is frequently used informally. Typically, this refers to how components of a design are connected to one another. The work of Reddy and Cagan (1995) provides a recent example of the value of formalizing such intuitive connectivity notions, as these authors use shape grammars as a basis for an algorithmic approach to optimizing truss design. In our opinion, a mathematics of engineering design will be significantly different from the traditional mathematics of engineering science.

The mathematical specialty of topology may serve as the foundation for any maturing mathematics of design, as a fundamental aspect of topology entails imposing a structure on certain subsets of a set, and properties of that structure may reflect conceptual aspects of the design or the design domain. In this paper, we review various research applications of topology to engineering design. At the end, we propose directions for future research that currently appear promising. Our goal is to present applications of topology to encourage other researchers to contribute to a scientific basis for engineering design. This is intended to stimulate further investigation into the role of topology as a unifying design abstraction.

A primary theme in selecting topics for this paper was unification. Consider the area of explanatory models of concurrent engineering knowledge and experience. The term General Design Theory (GDT) was coined by Yoshikawa (1981), who used topology to study the structure of abstract design concepts of functions, attributes, and their relationships. GDT was a motivating influence in the author's investigation of

Correspondence and offprint requests to: Dr D. Rosen, School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0405, USA. (e-mail: david.rosen@me.gatech.edu).

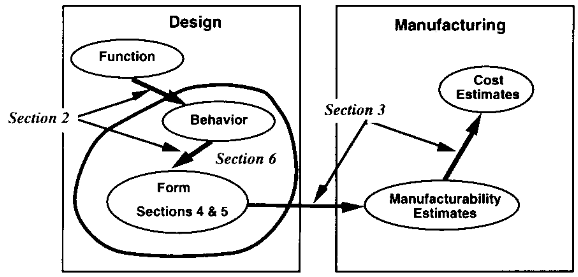


Fig. 1. Areas of application of topology in engineering design.

the nature of design–manufacturing conversions of engineering components. Another major area is geometric modeling. Topology is the underlying mathematical model for developments in mixed dimensional geometric modeling, tolerances, and modeling physical behavior. Thus, topology may serve as the unifying framework for theories, methods, and tools related to the representation of geometry, variations from nominal geometry, and behavior. Attempts to unify these two major areas has not yet been achieved, but we argue here that the abstraction of a topological perspective may provide the catalyst for such a unification of design theory.

Topology is a structured set of subsets of a given set (please see Appendix: Topology Fundamentals). A topological space is an ordered pair comprising the given set and a topology on that set. A well known topological space is the real line **R**. The standard topology imposed upon the real numbers to form the real line can be defined by all open intervals of reals. By specifying other topologies, the same set of real numbers can assume unusual properties. Furthermore, many formal properties of functions depend upon the topologies imposed upon their domains and ranges.

We attempt to focus attention on the engineering design applications of topology while simultaneously appreciating that an engineering audience[1] may not be readily conversant with the fundamental definitions of topology. Hence, a brief introduction to the fundamentals of topology is included as an appendix. Since this appendix does not presume to serve as a complete introduction to topology, the interested reader is referred to an introductory text such as Munkres (1975) or Willard (1970) for more details.

Our emphases for applications of topology in engineering design are summarized graphically in Fig. 1, which is explained as follows. Each ellipse represents a set of engineering information that is needed during design and manufacturing. Specifically, *form*, as shown in Fig. 1, denotes parameterized, feature-based geometric models, possibly with tolerances. Arrows indicate transformations between these information sets that we discuss in the next two sections. In Section 2, models of engineering design knowledge are presented using topological principles, where GDT is summarized. In this section, we also introduce a design example which is used consistently throughout the rest of the article to illustrate how the diverse topological techniques can be integrated within the design process. Our own work in characterizing the transformation of product information between life-cycle activities follows in Section 3. The focus shifts to the unification of solid modeling with more general geometric modeling environments, tolerances, and behavior models in Sections 4, 5, and 6, respectively. Our claim that topology may serve as a foundation for engineering design theory is supported by the broad coverage indicated in Fig. 1. In the final section, we present directions for future work.

2. Models of the Design Space

In this section, we will investigate research that defines types of topological spaces to describe relationships between problem (functional) specifications, design solutions, and intermediate behavior-based representations, as shown in Fig. 1. In effect, the work formally defines aspects of mechanical design knowledge via separation properties and metric spaces.

For a typical design problem, a set of candidate solutions are considered and each candidate will usually have unintended behaviors. By using metric spaces, the similarity (or difference) of design solutions can be measured in terms of the functions each performs, and the distance a particular solution is from the required functionality can also be measured.

2.1. General Design Theory

As a basis for understanding mechanical design activity, Yoshikawa (1981) proposed General Design Theory (GDT), which attempts to model the relationships between problem specifications and design solutions (such as machines) which are called entity concepts. His principle assumption is the Axiom of Operation which states that abstract concept sets are topologies (Appendix, Definition A.1) of the entity concept set. Entity concepts are idealized representations of existent or nonexistent entities (design solutions). Abstract concepts are function (intended behavior) and attributes. As an example,

[1] A complementary short survey article (Peters *et al.*, 1994*b*) was undertaken to introduce topologists to the applications of topology within Computer Aided Geometric Design.

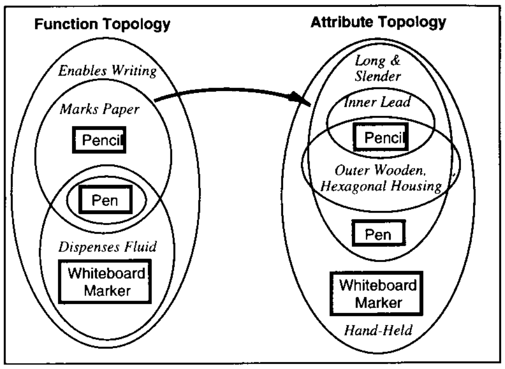


Fig. 2. Abstract function and attribute topologies.

pencils enable writing (function) and are long and slender (attributes) with an inner lead and an outer, wooden, hexagonal housing (more attributes). Hence "enables writing" is an open set in the function topology, while "long and slender" is an open set in the attribute topology. In Fig. 2, a function and an attribute topology are shown on the same set of writing instruments. Each circle represents an open set in one of the topologies and most are named by a function or set of attributes.

By defining a topology on entities, Yoshikawa defines the relationship between functional specifications and potential design solutions. That is, for each functional specification, there is a set of entities that can fulfill that specification (solutions to the problem). The set of entities that can fulfill a specification is an open set in the function topology. Entities that can fulfill two specifications are found by intersecting the sets of entities that meet each specification individually. The same type of relationship occurs with attributes.

With function and attribute topological spaces, design activity can be defined as a mapping from functional space to attribute space. That is, given a specification $t = (s_1, s_2, \ldots, s_n)$, there corresponds a design solution represented by m attributes, $d = (a_1, a_2, \ldots, a_m)$. Ideal knowledge is the knowledge of these topologies and the mapping between them. With ideal knowledge, design problems are solved once the problem specification is completed since the solutions are always contained in the intersection of open sets in the function and attribute topologies.

2.2. Metamodels

Tomiyama *et al.*, (1989) extended GDT to deal with real, as opposed to ideal, knowledge by addressing finiteness in the representation and in processing speed. They introduced another abstract concept, that of behavior based on physical laws, such as conservation of mass. The set of entities in this study was restricted to physically feasible entities, those that do not contradict known physical laws. Since behavior is another abstract concept, the set of behaviors defines a topology on the set of feasible entities.

Rather than treating design as a mapping from functions to attributes, Tomiyama *et al.* defined design activity as a stepwise, evolutionary transformation using concepts of behavior as intermediate states. A new topology of *metamodels* was defined on the entities, where open sets in the metamodel topology must be open in both the attribute and the behavior topology as well. In this way, the entities in an open set in the metamodel topology have a certain behavior in terms of physical phenomena and have the same attributes. This combination of information allows metamodels to be the bridge between functional specifications and design solutions.

Viewpoint specific models of design solutions are created from metamodels for analysis and other purposes through a knowledge of appropriate physical phenomena and the appropriate structure (or geometry) of solutions. Thus, from the theory, design methodologies and requirements for CAD systems, such as data representations, are derived. Kiriyama *et al.* (1991), extended the metamodel work and built a system for the design, simulation, and diagnosis of electric motors and their possible failure modes. The metamodel topology can be defined by taking all open sets from the attribute topology that are equivalent to an open set in the behavior topology.

2.3. Metric Spaces

Taura and Yoshikawa (1991) extended Yoshikawa's original ideas by distinguishing between function space and partial function space. Partial function space captures the functionality of individual components, rather than the system as a whole. Their motivation was to discover a design principle that is general, practical, and computable. Their hypothesis was that individual entities that perform similar functions (partial functions) can, when put together to form entire machines, perform similar total functions. By defining topological spaces for total and partial functions, they formalize the hypothesis and have a basis for testing it.

A metric (Definitions A.6 and A.7) was defined on the sets of entire machines to define the total function space. Similarly, a metric was defined on sets of individual components to form the partial function space. The metric for the total function space measures the similarity between machines based on the

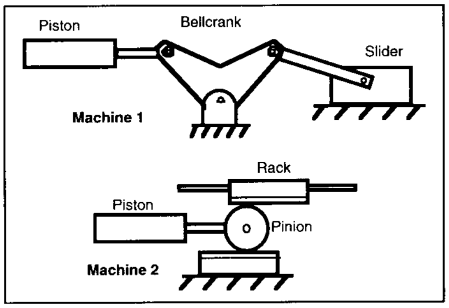


Fig. 3. Two machines with partial functions of (translate, rotate, translate).

difference in functions performed by the machines. Thus, machines sufficiently close to one another in the total function space perform very similar functions, while those machines that are far apart in the metric perform very different functions.

Based on their theoretical work, they developed a design methodology for searching in the total and partial function spaces and in the set of components. Their metric measures the similarity of function concepts and directs the search toward components that meet the required functionality. These components are put together into an assembly, analyzed, then evaluated to check that the assembly does, indeed, satisfy the functional requirements.

2.4. Kinematic Design Example

Taura and Yoshikawa (1991) developed an application to test their ideas in the kinematic design domain. An example problem might be to find machines that amplify rectilinear motions. As an example, consider two machines that amplify rectilinear motion and have the same partial function structure (translate, rotate, translate): a piston-cylinder/bellcrank/slider machine, and a piston-cylinder/pinion/rack machine (see Fig. 3). In this case, each component of both machines can be assigned a particular partial function. To seek improved designs, design operations can be carried out by changing partial functions; i.e., operating in partial function space. Partial functions can be added, modified, or deleted to yield new sets of designs. For example, the partial function structure (translate, rotate, translate) could be changed to (translate, rotate, rotate, translate) by adding a *rotate* function. Then their database of components can be searched and potentially improved machines can be generated. To determine whether or not these alternatives better fulfill the desired total function, each machine is analyzed. The idea is to modify the *partial functions* of known machines to achieve better total functions, under the assumption that similar partial functions typically yield similar total functions.

The bellcrank in Machine 1 of Fig. 3 will serve as the consistent design example in subsequent sections of this paper.

3. Models of Design–Manufacturing Conversions

In this section, we summarize some of our own work on the theoretical aspects of the design–manufacturing conversion of feature-based representations for the domain of thin-walled components. An explanatory design-manufacturing conversion theory was the goal of this project. Specifically, we investigate tooling cost evaluation of injection molded and die cast components. By defining the conversion of feature-based representations as a mathematical function, the properties of conversions can be investigated formally. Further, the study of properties of functions requires formal treatment of their domains and ranges. A topological space was constructed from the set of component representations in the design viewpoint. Similarly, a manufacturing topological space was defined on the set of tooling cost representations. Properties of conversions from one space to another are not only of theoretical interest, but should guide the development of methods and algorithms for product data conversion. As shown in Fig. 1 by the transitional arrow between the two rectangles, we are interested in estimating component manufacturability given a design representation of that component, then estimating its manufacturing cost.

As it is common knowledge that features useful for design are often distinct from features useful to manufacturing, it is necessary to convert feature-based component design representations to feature-based manufacturing representations (Cunningham and Dixon, 1988). Our design features for thin-walled components include flat walls, curved walls, holes, ribs, bosses, etc., which serve as the critical features within a prototype software system (Rosen, 1992). Design features are abstractions of geometry and contain size and location parameters. To design a component, the designer adds and combines features. For example, to combine two flat walls to form an L-bracket, the designer could use a rounded-corner feature. The component representation in our scheme then consists of one feature combination: (wall1, corner1, wall2). Design activity continues with the designer adding and combining features until the component is completed. Every feature combination is represented similarly as a triple. The configuration of a component refers to the feature combinations and feature types in the component.

A topological space was defined on the set of all feature-based design representaions of components based on a measure of the difference between components. Thus, a metric was defined on the design representations. Intuitively, two components differ if their configurations differ (different feature combinations or feature types) or, if their configurations are identical, while the values of size or location parameters differ. Thus, the difference between two L-brackets with bosses could be given by the difference in boss radii, assuming that all other aspects of these two L-brackets designs are identical. Defining a metric on the set of design representations induced a metric topology on that set (Appendix, Definition A.7). We denote the design topological space C.

Design-for-manufacturing (DFM) tools are often based on coding systems, which group components according to their similarity in processing. A 7-digit coding system and DFM tool have been developed for tooling cost evaluation in molding and casting (Poli *et al.*, 1988). Three of the digits correspond to *Shape* (*Flat* or *Box*), *Number of External Undercuts*, and *Amount of Cavity Detail*. Based on this DFM tool, undercuts and cavity detail were identified as manufacturing features (Rosen *et al.*, 1992). In Rosen and Peters (1992) it was shown that coding systems induce a topology on the set of manufacturing representations that is very different from the design (metric) topology. This work was subsequently further generalized (Peters *et al.*, 1994*c*). The topological essence is that there can exist points in the design space, arbitrarily close to one another, whose converted representations in the manufacturing space are very much separated from each other. The manufacturing space is denoted by M.

With topological spaces defined for design and manufacturing representations, it is possible to investigate the mapping from one space to the other. In particular, we are interested in $f\colon C \to M$, the mapping from design to manufacturing. Since one of the fundamental properties of mapping is continuity, the continuity of $f\colon C \to M$ is of interest. Definition A.11 and Theorem A.12 present some important properties of continuity. Within the domain of injection molded parts, it can be shown that this mapping can have discontinuities, some of which arise from the physics of the manufacturing process, while others are due to the nature of the manufacturing topology, induced from the coding system.

To illustrate an example discontinuity that arises from the manufacturing process, consider the bellcrank

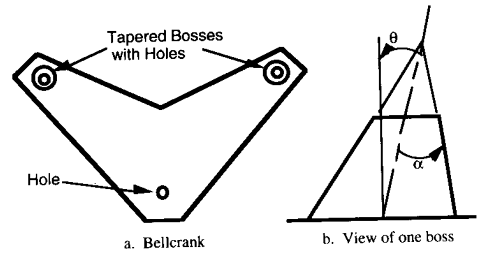


Fig. 4. Bellcrank consisting of two tapered bosses and a hole on a flat wall.

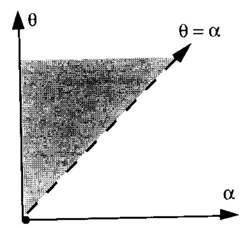


Fig. 5. Tilt angle versus taper angle showing lines of discontinuity.

from Fig. 3, now shown in more detail in Fig. 4a. We will focus on one tapered boss. Let θ be the measure of the boss tilt angle and α be the measure of taper angle, measured relative to θ. The view in Fig. 4b corresponds to a plane that contains the boss axis and is perpendicular to the flat wall. When the axis of the boss is parallel to the mold closure direction, the boss contributes to cavity detail.[2] However, if the boss tilts sufficiently so that tilt is greater than taper ($\theta > \alpha$), then the boss causes an undercut and does not contribute to cavity detail. Since undercuts require side-action units in the tooling, tools for components that contain undercuts are more expensive than tools for simpler components. Thus, the taper angle versus tilt angle curve has a line discontinuity along $\theta = \alpha$ as shown[3] in Fig. 5. The shaded part of the graph denotes sections of the design space that correspond to the presence of undercuts. Near the discontinuity, a small change in design can cause large changes in manufacturability. As shown here, a small variation in tilt angle, an orientation parameter, or taper, a design parameter, can cause large change in tooling cost. In relation to the topological spaces, this discontinuity appears when infinitesimally close neighbors in the design space are mapped to different open sets in the manufacturing space as shown schematically in Fig. 6. In the manufacturing space,

[2] Note that α cannot be negative, as that would imply a reverse taper.

[3] More correctly, some tilt beyond $\theta = \alpha$ can be tolerated in practice. The criterion for an undercut would become $\theta = \alpha + c$ where c is an empirically determined angle, dependent upon processing conditions. See Peters *et al.* (1994*c*) for a more thorough analysis.

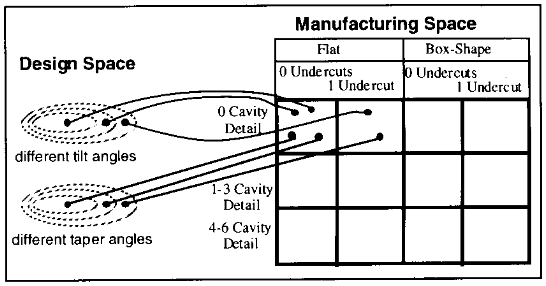


Fig. 6. Mapping from design to manufacturing illustrating discontinuities.

each box represents an infinite number of parts, all of which share the same manufacturing characteristics. For example, all parts in the upper right corner are box-shaped, have one undercut and have no cavity detail. The discontinuity illustrated in Fig. 6 shows a small change in boss angle tilt causing a jump from the manufacturing category represented by

$$\langle \text{Flat, 0 Undercuts, 0 Cavity Detail} \rangle$$

to the category

$$\langle \text{Flat, 1 Undercut, 0 Cavity Detail} \rangle.$$

A potential critisism of this example is that it is very process specific. Three questions can be asked:

1 Can another manufacturing topology be formulated that would result in continuous conversion functions?
2 What does the investigation methodology say about the knowledge requirements for concurrent engineering?
3 Does the investigation methodology generalize to other manufacturing processes?

Addressing the questions in the order asked, recall that the manufacturing topology was generated from a group technology coding system which separates parts into discrete categories. As such, this type of coding system will always have the discrete topology (see the discussion after Definition A.2 in the Appendix). In turn, this implies that the topology induced on the set of manufacturing representations is extremally disconnected, as each open set is also closed (Definition A.14). Intuitively, a discontinuity arises because two points within a small open set of the design space may be mapped individually into very separate open sets of the manufacturing space.

Regarding the knowledge requirements for concurrent engineering, it is of interest to investigate whether other topologies could support continuous design-manufacturing transformation functions. For instance, first consider the transformation function as a mapping purely between the underlying sets, with no topologies yet imposed. Then, define a topology on the design space as the weak topology (Definition A.13) specified by defining open sets to be the inverse images of open sets of the manufacturing space. While this necessarily guarantees continuity, this weak topology implies full knowledge of the manufacturing space before the design is finalized, a situation that rarely occurs in practice.

It is notable that emerging "rapid prototyping" processes, and some existing processes such as wire-cut EDM (electrical discharge machining), have a fixed tooling cost and a variable processing cost that depends primarily upon the part length, volume, or perimeter measures. These measures vary continuously, so initial analysis might suggest that conversion from a design space to these manufacturing spaces may be continuous as a result. However, some subtleties do arise in these considerations, as expressed in Peters *et al.* (1994*c*).

Several generalizations are possible. First of all, this type of analysis applies to any representation transformation, provided topologies are specified on the domains and ranges. In particular, topologies are easily defined for life-cycle functions that are describable by coding systems. Second, these transformations can have discontinuities. In the examples presented, these discontinuities help explain why small design changes can have large impacts on manufacturing. Third, the discontinuities help explain why manufacturability evaluations can be difficult. Fourth, we note that the design topology was specified without information from the manufacturing viewpoint. We conjecture that the incorporation of manufacturing knowledge into design activity, as is desirable in concurrent engineering pratices, can be modeled as modifications to the design space topology to reflect the design knowledge. Lastly, some emerging and existing manufacturing processes may possess characteristics that ensure continuous conversion functions.

4. CAGD Models of Mechanical Components

In order to provide computational tools that effectively support the design process, it is of considerable research importance to focus upon the development of unified, general geometric modeling environments. In this section, topology is presented as the unifying formal basis for both solid and non-manifold modeling. Computer aided geometric design (CAGD) has evolved from an initial emphasis upon geometric concerns to a more mature focus upon topological issues.[4] As geometric modeling is typically assumed to be undertaken within Euclidean spaces, the topology understood within this section will be the usual Euclidean metric topology (Definition A.6).

For any early discussion of solid modeling issues and representations, see Requicha (1980), while Requicha and Voelcker (1982) put solid modeling into an historical perspective. Mäntylä (1988) offers an accessible introductory textbook on solid modeling, along with data structures and code. Within CAGD, topology concentrates upon adjacency relations amongst vertices, edges and faces, where the Euler–Poincaré Equation provides a classical defining relationship. Note, that this Euler–Poincaré Equation is an algebraic equation used to summarize important topological information. This topology/algebra interplay will appear as an important underlying theme and this particular occurrence is notable as the first instance of it.

4.1. Solids as Regular, Closed Point-Sets

Consider a unit cube with a vertex at the origin and extents in the positive X, Y, and Z directions. Then, from a point-set perspective, every point in space can be classified as in, on, or outside of this cube by determining if the x, y, and z coordinates of the point are between 0 and 1, inclusive. The well known set operations of union, intersection, and difference apply naturally to objects given by point sets.

Solid modeling is a prominent speciality within computer aided geometric design. The proliferation of inexpensive graphics workstations allows solid modeling to be an integral aspect of CAGD modeling. Solid modeling may be considered as a system of operands and operators. The solid operands are required to enclose positive finite volumes.

An intuitive concept of a solid is a closed, bounded, rigid point-set that has a homogeneous boundary; i.e., no dangling edges or faces and no isolated vertices. In this context, closed means that the object contains its boundary, while bounded means that the object has a finite extent. While this intuitive understanding should be sufficient for comprehension of this paper, the open sets and closed sets of a space can be formally defined (Munkres, 1975) once a topology has been specified upon the space. For one dimensional Euclidean space, the non-inclusive interval (0, 1) is a representative open set, whereas the inclusive interval [0, 1] is the corresponding representative closed set.

[4] Even more abstractly, some research is extending the topological formalisms into purely algebraic formalisms (Shapiro, 1991).

The requirement for a homogeneous boundary can be formalized with the concept of regular sets. Informally, to make a point-set regular, one first strips off all boundary points including dangling edges and faces, then covers the result with a "tight skin" of bounding points. More formally, for a point-set A, the regularized point-set $r(A)$ is given by

$$r(A) = cl(int(A)) \tag{1}$$

where $int(A)$ denotes the interior of A and cl indicates the closure operator which contains this interior with a minimal boundary. These topological operators of interior and closure (Munkres, 1975) easily follow upon the formal definitions of open and closed sets, respectively.

One technique for building more sophisticated solid models utilizes binary combinatorial operators upon primitive solid operands, such as spheres, tori, blocks, cylinders and wedges. Initially, the solid operands were conceived primarily as point sets, and the necessary binary operators were seen as set union, intersection and subtraction. However, the intent was that the resultant of any such binary operation would be a valid solid and, thus, could be considered as an operand for another binary operation. The set intersection operation formed the basic kernel of algorithms for all three operations and it was readily apparent that the set theoretic intersection operator could easily yield resultants which were not valid solids (Tilove and Requicha, 1980) (e.g., consider two closed unit cubes aligned so that their intersection is merely a planar unit square, which is not a solid because it clearly fails to contain a positive finite volume). The Boolean algebra of regular closed sets was seen as a formal expression of the necessary binary operators. This Boolean algebra on regular closed sets is a prominent example of the topology/algebra interplay previously mentioned.

As a result, for solids A and B, the intent of *intersection* and *union* were replaced, respectively, by the following operations,

$$cl(int(A \cap B)), \tag{2}$$

$$cl(int(A \cup B)),$$

and the *subtraction* of B from A can be specified notationally as

$$A - B = cl(int(A \cap B')) \tag{3}$$

where B' indicates the set complementation of B. These three operators are collectively referred to as *Booleans*. From a mathematical perspective, these operators are instances of *meet*, *join* and *complementation*, respectively, where the first two terms derive from a Boolean algebra being a particular type of lattice. However, the original terminology persists, and these three operators are referred to individually, albeit somewhat imprecisely, as union, intersection and subtraction.

For practical computation of any of these three binary operations, the intersection between the two operands must be evaluated. Hence some special pathological cases relative to surface intersection algorithms merit discussion. These are collectively known as degeneracies and involve very small intersections or tangencies. Under these circumstances, careful attention must be paid to the integration of the floating point computations (upon the underlying geometric representations) with the topological reasoning (upon the interrelated design objects), so as to prevent gross, unintended topological results, possibly leading even to invalid solids. This is an issue inherent in the use of a numerical algorithm of finite precision and is the subject of considerable research.

A research subject is to identify types of intersections that cause such pathological cases and to process them gracefully, creating a happy marriage between the numeric implementation and the required topology. Hence, topological and geometrical properties of solid objects should be used to guide the computation of Boolean operations. While many CAGD systems have developed *ad hoc* "software patches" to process specific known cases of these degeneracies, these attempts can be inconsistent and/or misguided. Proper attention to a rigorous theoretical topological foundation can facilitate stable CAGD software implementation. Historically, Weiler (1986, 1988) offers evidence as to the value of such an approach. Furthermore, such formal foundations are likely to be of even greater importance as future CAGD systems will expand to address increasingly varied and complex geometric and topological domains.

To indicate a practical implication of the concern for consistent model topology, note that such software inconsistencies in CAGD systems can cause problems downstream. The use of stereolithography and other "rapid prototyping" technologies demonstrates that attention to the theoretical aspects of maintaining valid solids can have very practical implications, as evidenced by the recent work of Bøhn and Wozny (1993). The premise of their article is that models that do not adhere to the topological requirements of regular closed sets described above "... frequently cause problems during fabrication". The standard data representation for stereolithography approximates the boundaries of solids by triangles. As a consequence of the imprecision of floating point arithmetic, this approximating geometry may fail to faithfully represent the topology of the object. Representative problems would be the creation of disconnections between faces that were previously connected or reversal of facet normals leading to surface orientation difficulties. Such topological inconsistencies can result in significant difficulties for faithful model creation by means of stereolithography. The rest of their paper describes algorithms and heuristics to remedy these topological deficiencies so as to allow the manufacturing cycle to proceed. Their conclusion states "The solution is numerically robust and it is based on topological principals [sic]".

4.2. Non-Manifold Solids

The previous section introduced the role of Boolean algebras within CAGD, continuing the theme of topology/algebra interplay. For solid modeling, the space over which a Boolean algebra is considered is $\mathbf{R}^3$. The concern over retaining bounded volumes within $\mathbf{R}^3$ motivated the further restriction that all solids must also be semi-analytic sets, resulting in further specialization of the algebra for combining the modeling operands. The knowledgeable reader is aware that regular closed sets need not be restricted to $\mathbf{R}^3$. In fact, CAGD applications within $\mathbf{R}$ and $\mathbf{R}^2$ have already proven quite useful. The need for modeling in $\mathbf{R}^n$, where n is a fixed, but arbitrary positive integer, is gaining increasing importance. However, several of the assumptions made about valid resultants enclosing finite volumes restricted the initial versions of solid modeling code from being gracefully integrated with software for modeling lower and higher dimensional geometry.

The seminal work on providing generalized data structures, algorithms, and a software engineering basis[5] for that integration was performed by Weiler (1986, 1988)[6] under his terminology of *non-manifold topology*.[7] While recognizing the intimate relationship between CAGD topology constructs and the underlying computational geometry, the work of Weiler (1986, 1988, 1992) was a catalyst for computational topology

[5] While software engineering issues are *not* the focus of this article, it is worthwhile to note that K. J. Weiler utilizes an edge-based representation, where each edge serves as a reference from which adjacency relations are determined. Other practitioners (Gursoz *et al.*, 1988) prefer a vertex based approach.

[6] The integration of surface and solid modeling techniques was also undertaken independently in the early 1980s at the University of Utah (Alpha, 1990).

[7] Recent discussions within the CAGD community (IFIP 92, 1992) have raised some concerns about the lack of precision in the use of the term "non-manifold topology" and the expression "mixed-dimensional modeling" has been advocated as more suggestive.

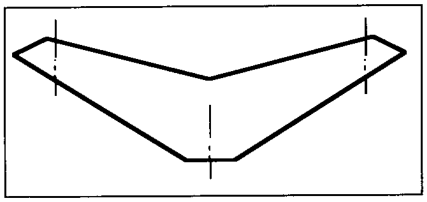


Fig. 7. Non-manifold model of the bellcrank utilizing a geometric abstraction.

to achieve a mature software engineering role as an independent subsystem. He developed geometry and topology subsystems independently, explaining his motivation as follows (Weiler, 1988):

> "Use of topological properties can simplify modeling algorithms and greatly improve their efficiency. However, for several reasons, topology can be even more useful when it serves as the framework around which the geometric modeling representation can be built."

The major legacy of this work has been "... to eliminate the formerly large conceptual gaps between the different dimensional element modeling techniques and provide a uniform environment for these techniques in both a conceptual and practical sense" (Weiler, 1992). Previously, within the CAGD community, the conventional wisdom required sharp boundaries separating curve, surface and solid modeling techniques from each other. CAGD researchers "... concentrated on ... [these] differences rather than their unification" (Weiler, 1992). It was Weiler's observation that human designers "... use a continuum of related geometric entities in a single domain rather than separate geometric entity sets in separate domains" (Weiler, 1992).

For example, as has already been noted, solid models are desirable in many design activities. However, the modeling effort to describe solid objects fully creates additional conceptual and computational demands. For instance, consider the bellcrank from Figs 3 and 4a. Early in the design of motion transmission machines, geometric details are not as important as the motion characteristics of the machines. As such, geometric abstractions can successfully convey overall part shape and motion characteristics to the designer. In Fig. 7, a geometric abstraction is used that ignores part dimensions on the order of wall thicknesses. The resulting non-manifold model of the bellcrank consists of a mid-plane to represent the body and center-lines to represent the bosses and the holes. One advantage of non-manifold modeling is that it allows solids, surfaces, curves and vertices to co-exist as modeling primitives within the same software environment. This saves time and energy at the conceptual stage, as well as being far more computationally efficient, both with respect to storage space and CPU time. This conceptual model could then serve as the basis for a full solid model at a later stage for accurate shape description, mass properties calculation, and interference detection. Using a non-manifold modeler, such as Noodles (Gursoz *et al.*, 1988), the designer can selectively refine shape details, for example one tapered boss, without having to use exclusively 3-D geometric primitives throughout the bellcrank.

Once the usefulness of such mixed dimensional geometric techniques became apparent, unification, rather than differentiation, was pursued. The topological abstractions were sufficiently general to provide a means of unification over differing geometric representations. In particular, such topological abstractions were fundamental to recent work (Desaulniers and Stewart, 1992) addressing formal transitions from manifold solids to non-manifold regular closed sets. Other engineering research applications of non-manifold modeling have spanned feature recognition (Gadh *et al.*, 1991), feature-based design (Rosen *et al.*, 1992) and several types of geometric abstractions for reasoning about shape (Zamanian *et al.*, 1992; Sudhalkar *et al.*, 1993).

4.3. Unified Geometric Modeling

The recent doctoral dissertation of Shapiro (1991) may be seen as an extended exploration of the topological/algebraic interplay within the broad realm of geometric modeling. Geometric modeling also includes other non-solid operand/operator schemes, utilizing differing supporting topological abstractions. Shapiro seeks a unified algebraic formalism for all such CAGD modeling schemes. He is specifically motivated in his search by his realization that in current CAGD "... the lack of a common formal language leads to the proliferation of informal concepts and results in many redundant efforts that are often ill-defined" (Shapiro, 1991).

His approach is based upon complementary topological and algebraic analyses of the formalism of geometric design. Shapiro investigated commonalities among representation conversion processes, such as CSG $\rightarrow$ BRep, BRep $\rightarrow$ CSG, and CSG representation optimization. The ability to perform representation conversions efficiently is essential since different applications (finite element analysis versus design) often require different representations. The key is to investigate space decompositions of objects and to associate specific types of decompositions with specific types of representations. For each type of representation, he proposes the establishment of a canonical format. Furthermore, a *neutral canonical format* is proposed through which any representation type can be converted. Then, the topology/algebra interplay is seen as the formal framework for the unification of geometric modeling.

5. Models of Geometric Tolerancing

Tolerances have been used in mechanical engineering to bridge the gap between design and manufacturing, since tolerances express allowable deviations from nominal form that can be produced during manufacturing. Without consistent, computable, and measurable tolerance models, efficient automation and integration of life-cycle activities is prevented (Juster, 1992). From a design viewpoint, tolerances are crucial in specifying allowable degradations of product performance. Robust design is the practice of relating tolerance values to overall measures of quality. In this section, we review tolerance research whose formal basis may provide sufficient abstraction to begin to bridge the gaps between design, manufacturing, and the customer's perceptions of quality. As in the previous section, the underlying topology of the modeling domain is understood to be the usual Euclidean metric topology.

Formal consideration of tolerance issues has been strongly motivated by the alarm cited in the "metrology crisis" (Voelcker, 1990, 1991, 1993). Indeed, it was observed (Voelcker, 1991) that the American tolerancing standard (ANSI, 1982) is "... dangerously inadequate because it is cast in prose and special-case examples, and does not provide the precise mathematical definitions that are required to write control programs for inspection machines". In particular, lack of formal, consistent, software calibrations of coordinate measuring machines had significant economic impact. As Voelcker states (Voelcker, 1990):

> "For several weeks the flow of mainly military hardware worth hundreds of millions (possibly billions) of dollars simply stopped, because work could not proceed until the goods could be dimensionally qualified. *Ad hoc* agreements between DoD authorities and standards organizations got the flow restarted, but the underlying problem(s) remain"

Similarly, for algorithms for non-contact metrology, Besl (1994) has observed that the problem of checking for correct topological form is a serious unresolved difficulty.

The role of topology within tolerance theory was implicitly recognized as long ago as 1983 by Requicha (1983), as he articulated the need for a formal computational model for geometric tolerances. In a recent survey, Juster (1992) stresses that the success of modern manufacturing industries is dependent upon computer modeling of tolerances for the reasons presented above. He cites some modest advances, but emphasizes the immaturity of the field.

As expressed in his seminal article (Requicha, 1993), there exists two guiding principles that merit consideration in developing a formal tolerance theory:

1. It should be possible to construct a neighbourhood about the boundary of the nominal object in which geometric variations would be permitted.
2. The form of the objects within the tolerance must also be constrained to be similar to that of the nominal object.

In that work, these two principles were expressed intuitively and illustrated by examples. The first principle corresponds to the notion of a tolerance zone. The second addressed that all possible geometric variations wihin that zone are not equally acceptable. For instance, to utilize one of Requicha's examples, consider placing tolerance values upon the radius of a circle. Then the Euclidean metric neighbourhood, corresponding to its tolerance zone, would be an annulus, whose inner radius would be the minimally allowed circle radius and whose outer radius would be the maximally allowed circle radius. However, all geometric variations within that annulus would not be equally acceptable. An ellipse might be, but any closed curve with a self-intersection would most certainly not be, because the original circle had no such self-intersections.

While this brief introduction offered here is still mostly informal, Boyer and Stewart (1991, 1992) and Stewart (1993), formally introduce sets of objects which would be admissible within a tolerance zone. In a recent survey article, "Solid Modeling and Beyond" (Requicha and Rossignac, 1992), Boyer and Stewart are lauded as follows:

> "... we see progress in the mathematics of tolerance specifications, which also define sets of sets, called variational classes.... Recent work by Boyer and Stewart at the University of Montreal proposes sharp mathematical characterizations for variational classes in terms of regularity in a topology related to the Hausdorff distance between sets."

The initial Boyer and Stewart work (1991) refines two fundamental topological tools as a basis for their mathematical characterizations

1 a specially defined metric (Definition A.6),
2 a specialized type of homeomorphism (Definition A.8).

Their specially defined metric[8] ensures that a set "within tolerance" of the nominal object is sufficiently close to the nominal object. The associated metric topology leads to the definition of a Boolean algebra of regular closed sets, and is another manifestation of the pervasiveness of the interplay of topology and algebra. Both the metric and the algebra are crucial to the formalisms they develop. However, these techniques are augmented by their second tool, a specialized type of homeomorphism.

Even when two sets have their metric difference being sufficiently small, those authors proposed additional requirements for preservation of the topological characteristics between the nominal model and its variants. They persuasively argue, using their example of the knotted torus, that the traditional mathematical approach of establishing topological equivalence merely by homeomorphisms is not sufficient for tolerance modeling. Their specialized homeomorphisms, known as "space homeomorphisms", would distinguish between the standard torus and the knotted torus,[9] whereas mere homeomorphisms would not be able to do so. For a homeomorphism of two subsets of $\mathbf{R}^n$ to be a "space homeomorphism" it must be extensible to a homeomorphism of all of $\mathbf{R}^n$ onto $\mathbf{R}^n$. The knot in one torus precludes such an extension of any homeomorphism between it and the standard torus.

Past engineering practice has not necessitated a formalization of this same form notion, since two human experts could agree when the condition failed to be met. Future advances in engineering design will require a precise definition of when two objects or sets should have the same topological form and this has now been provided by the works of Boyer and Stewart (1991, 1992). Sufficient conditions for satisfying these definitions were given by Stewart for an important special polyhedral modeling case (Stewart, 1993). For the general class of polyhedral models, analogous sufficient conditions have been developed (Andersson *et al.*, 1994).

Consider geometric tolerances applied to the tapered bosses of the bellcrank from Figs 3 and 4. Application of the Boyer–Stewart tolerance model to the bellcrank bosses would specify conditions on boss and hole tilt, for example, that prevents a hole from piercing the tapered surface of a boss or that prevents a hole from disappearing. In other words, although the geometry of the bounding surface of the bellcrank may change, the topology of this bounding surface must not be changed by variations from nominal form. Algorithms to check those topological conditions have been incorporated into a demonstration software system (Dorney, 1994).

Contemporaneously with such investigations of the theoretical foundations for tolerance, the American Society of Mechanical Engineers (ASME) has prepared a new standards document for tolerancing. This standards document has not yet embraced the topological aspects relative to tolerance, as indicated above. However, the underlying topological concerns still appear to be as important as when originally raised by Requicha. A recent two-part special issue of *Manufacturing Review* on "Tolerancing & Metrology" echoes the spirit of the ASME standards effort, apparently being devoid of explicit consideration of the research of Boyer and Stewart. Rather, for the specific problem of classifying tolerance zones, the emphasis of the special issue was upon using offset zones and geometric sweeps to define tolerance zones (Requicha, 1993; Srinivasan, 1993). These offsets and sweeps are primarily geometric techniques, whereas the approach of Boyer and Stewart was principally topological. It appears that tolerance theory needs the complementary influences of both geometry and topology. Hopefully, both the ASME and the topological approaches can be synergistically combined to yield a viable basis, simultaneously pragmatic and rigorous, for tolerance.

6. Models of Physical Behaviour

In previous sections, we have indicated that once a topology has been established upon a set, this leads naturally to various algebraic characterizations of the underlying topological properties. This topology/algebra interplay has a rich mathematical history leading to the mathematical specialty of algebraic

---

[8] For those already familiar with the Hausdorff metric, it is noted that their specialized metric is a modification of the Hausdorff metric. However, knowledge of these mathematical details is not crucial to understanding the remaining presentation of this article and those details have been omitted for the sake of brevity. The interesting reader is referred to the Boyer and Stewart work (1991) for a clear, complete presentation of these details

[9] The original work (Boyer and Stewart, 1991) labeled these specialized homeomorphisms as "tame homeomorphisms", but later usage (Andersson *et al.*, 1994) has replaced this initial terminology by "space homeomorphism" and we will consistently adhere to this more recent usage.

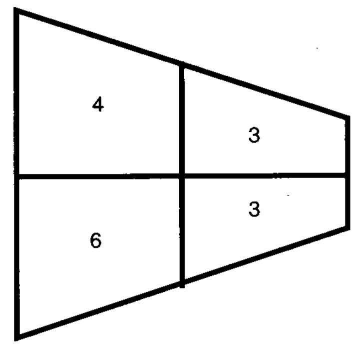
topology. Our preceeding use of the term *topology* referred more properly to the specialty of point-set topology. The discipline of point-set topology forms the basis for further specialization into the sub-disciplines of algebra topology, differential topology and geometric topology. This section will illustrate how the underlying Euclidean metric spaces can be augmented with notions from algebraic topology to provide an enhanced design theory model. This brief discussion will concentrate upon intuitively accessible, illustrative examples. The richness and rigor of the underlying algebraic topology theory is available in several standard texts, including Massey (1967) and Hocking and Young (1961). Additionally, the more recent article (Lear, 1992) offers some well articulated perspectives upon fundamental relations of algebraic topology to geometric modeling. The interested reader is referred to these sources for additional background.

In this section, a model of physical behavior is presented that extends the mathematics of geometric modeling, as illustrated in Fig. 1 by the combination of behavior and form. As such, it contributes to an emerging theoretical framework for product representations. Palmer and Shapiro (1993) claim that the key to achieving a computational model of physical quantities is to effectively compute with distributions of physical quantities in time and space. They seek to model physical quantities with finite representations in much the same way as finite elements are used as a representation for engineering analysis upon model geometry. As motivation for their work, they assert:

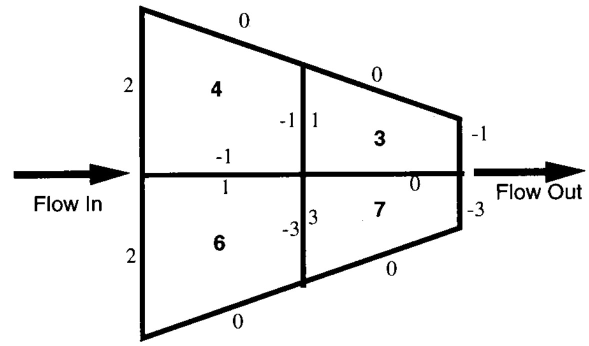
> "The relationship between geometry (form) and physical behavior (function) dominates many engineering activities. The lack of uniform and rigorous computational models for this relationship has resulted in a plethora of inconsistent (and thus usually incompatible) computer aided design tools and systems. It seems clear that formalization of this relationship is a prerequisite to taking full advantage of computers in automating design and analysis of engineering systems."

6.1. Cell Complexes

Palmer and Shapiro propose a model derived from algebraic topology based on cell complexes, chains, and topological operations on chains. *Cell complexes* are much like finite elements in that the geometry of an object is decomposed into a finite number of "cells". Corresponding to each cell is a distribution of a physical quantity represented by a



a) chain representing mass distribution.



b) chains representing fluid mass and fluid flow.

Fig. 8. Example chains representing mass distributions.

chain which, in this context, is a mapping from a cell complex K to a vector space G, representing a physical quantity such as mass, momentum, or charge. Thus, a chain is a map from $K \rightarrow G$. Consider the object in Fig. 8a consisting of four quadrilaterals. Each quadrilateral is a 2-cell that is bounded by 1-cells (edges) which are bounded by 0-cells (vertices). The numbers in the 2-cells are coefficients of a chain that represent (for example) the mass of the cells. Thus, the chain represents the distribution of mass in the object. Palmer and Shapiro refer to their development as *chain models of physical behavior.*

A second example is shown in Fig. 8b which shows identical cells but models fluid flow through the cells. Two chains are needed to describe fluid flow: one that models the amount of fluid in each 2-cell and one that models the fluid flow across the boundaries of each 2-cell. Each boldface number (e.g., **4**) denotes fluid in a cell, while the other numbers denote flow across a cell boundary. Positive flows are flows into a cell, while negative flows are outflows. Note that a cell boundary of 0 denotes either a barrier or flow solely along that boundary. By the conservation of mass law, the rate of change of mass in a 2-cell is equal to the mass flux through its boundary. This can be expressed by the following relationship between chains:

$$\frac{d}{dt} M(K) = -\delta(M_f(K)) \tag{4}$$

where $M(K)$ is the chain representing mass in the 2-cells, $M_f(K)$ is the mass flux through the boundaries of these 2-cells, and K is the complex of cells. The operator δ is the coboundary operator that associates with a cell the oriented sum of its bounding faces in the chain (see Palmer and Shapiro, 1993 for details). It is noteworthy that most physical laws can be modeled by topological relationships similar to Eq. (4).

6.2. Chain Models

Chain models are neither lumped-parameter models nor continuum models. In Shapiro and Voelcker (1989), the need to represent distributions of geometry and physical quantities was raised. They compared bond graph formulations of lumped-parameter dynamic models to a hypothetical model that represents varying levels of distribution. While recognizing the value of continuum models, Palmer and Shapiro argue that consideration of cell complexes can offer more effective computation since:

- continuous functions modeling physical quantities do not completely characterize system behavior until initial or boundary conditions are specified. Equations constrain behavior, but do not determine it.
- most differential equations of interest cannot be solved analytically.

They have embodied these principles within a research software system (Palmer, 1995).

If distributions of physical quantities are modeled by chains, then physical laws can be modeled as constraints on the coefficients of chains. In order to formulate a physical law as a set of constraints on chains, it must be possible to formulate the law locally; i.e., formulate laws by imposing constraints on adjacent or incident cells only. Palmer and Shapiro claim that most physical laws can be so formulated and divide laws into two classes: structural (e.g., conservation, equilibrium) and constitutive (e.g., Ohm's and Hooke's). Structural laws are purely topological statements that apply to all cells in all decompositions (as in Eq. (4)), while constitutive laws depend on the metric of the embedded space as well as on the cell types.

Physical elements are defined as cell complexes generated by a single cell of a particular type, with a set of chains that represent physical state and a set of constraints that embody the allowable changes in state. An example physical element was derived to represent elastic solid behavior and to illustrate these concepts. The chain that represents elastic solid behavior relates the behavior of the cell to the behaviors of its bounding faces, which, in turn, are related to the behaviors of their bounding edges, and subsequently to the vertices that bound the edges. A physical element that is nominally a cube is represented by 36 equations.

6.3. Relationship to Design

Chain models may lead to engineering design features with embedded geometry and physical behaviors. By associating chains that represent specific behaviors with common engineering shapes (that is, form features (Shah, 1991)) that are composed of cell complexes, useful design primitives can be developed. Furthermore, these design primitives can be available to both the designer and to automated design systems, for example, for shape optimization. Palmer and Shapiro also describe how chain models can be applied for engineering analysis and simulation. Physical elements can be thought of as objects from an "object-oriented" viewpoint in the sense that these elements encapsulate data and interact with other objects through well-defined interfaces (Peters *et al.*, 1994*a*). After an element type, or class, has been defined, it may be used to model any object that is intended to have its behavior, and may be combined with other element types to define composite behaviors.

If we think of engineering design as the development of physical systems to meet required behavior, satisfy constraints, and connect with their environment, then physical elements may serve as appropriate design primitives for physical artifacts. Shapiro and Voelcker (1989) postulate that mechanical functions can be modeled in terms of spatially distributed energy exchanges between parts in a system and its environment. While bond graphs and other lumped parameter representations can model energy exchanges, they cannot represent the physical boundaries over which these energy exchanges occur. Through the application of algebraic topology to physical behavior, Palmer and Shapiro's physical elements can represent and compute with physical boundaries.

6.4. Design Example

In their article, Palmer and Shapiro illustrate the role of the chain model theory in the design of a family of 3-hole, angular brackets, similar to the bellcrank example from Figs 3 and 4. Their design specification

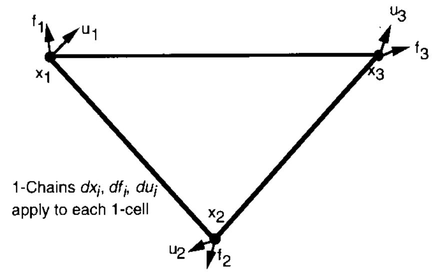


Fig. 9. Initial bellcrank geometry and chains.

is expressed as a set of physical constraints on chains. Then the design goal is to find a cell complex and a set of chains that satisfy these constraints, where it is understood that this complex is required to be embedded properly in Euclidean space of appropriate dimension. In the bellcrank design problem, the bellcrank transfers a load from the piston to the slider while rotating around a shaft connected to ground. The design specification for the bellcrank can be defined as a chain model that specifies relationships between hole loads and displacements.

Initial bellcrank geometry can be given as a "stick figure" with lines (1-cells) connecting each pair of holes, which are represented by nodes (0-cells). An initial chain model can be created by applying the chain model of the specification to this initial geometry as in Fig. 9. This is described by Palmer and Shapiro:

> "Each hole is represented by a 0-cell (a node), giving rise to a number of 0-chains: hole positions x, forces f, displacements of holes u. . . . Using the coboundary operation, we induce three 1-chains: relative position $dx = \delta(x)$, relative displacement $du = \delta(u)$, and relative force $df = \delta(f)$."

As the design process evolves, the chain model of the bracket evolves with the addition of higher dimensional physical elements to the bracket's underlying cell complex; e.g., replacing the 1-cells with a 2-dimensional version of the elastic solid physical element mentioned in Section 6.2, resulting in a geometric model similar to that shown in Fig. 7. Notice that the underlying representation and computational models do not change; everything is a chain model. Different design specifications embodied as different constraints on chains will result in different cell complexes, hence different geometric shapes of the bellcranks.

7. Conclusions and Directions for Future Work

In this paper, we have investigated the development of a mathematics for engineering design. Topology was suggested as a body of mathematics that could unify diverse areas of CAGD and engineering design research. Several examples of the application of topology to CAGD and design were presented. In particular, we surveyed applications of topology in modeling design spaces, modeling design–manufacturing information conversion, solid and non-manifold geometric modeling, geometric tolerancing, and modeling physical behavior. Since topology is fundamental for much of modern mathematics, it should be expected to appear prominently in engineering design mathematics. Unification of diverse design research areas relies upon a mathematical core that can span these areas; topology may serve as that core.

Several open questions of continuing research interest will be mentioned here. Note that in Fig. 1 the tight coupling among the areas of Solid Geometric Modeling, Tolerance Modeling, and Form reflects a desirable state, not the state of unification of underlying theories.

- Regarding the work of Palmer and Shapiro (1993), it might be desirable to develop a formal definition of features, as categorized by the physical elements of a design, and then test the representational and analysis advantages/disadvantages of these design features in a particular domain where the physics is well understood.
- It would be of interest to integrate tolerance modeling with design features, to capture not only physical behavior, but variations in that behavior.
- We hypothesize that topology provides a sufficiently broad, yet useful, abstraction to unify formal geometric and physical models within the research on conversions of product representations between life-cycle viewpoints, including physical behavior and tolerances. Might this require extending conversion models from a single-objective formulation to multi-objective?
- A taxonomy of engineering design problems has so far proven elusive. It may be that a taxonomy can be formulated based upon the types of topological spaces under consideration. For example, parametric design (the assignment of values to design variables) is performed in a subset of $\mathbf{R}^n$, while configuration design (the elements of a design, their connectivity, and approximate positions and sizes) is performed in sets of subsets of $\mathbf{R}^n$.
- A diversity of software systems for modeling topological design considerations have been reviewed (Gursoz *et al.*, 1988; Kiriyama *et al.*, 1991; Rosen, 1992; Dorney, 1994; Palmer, 1995). We hope that

this article will serve as a catalyst for broad software integration of such emerging tools. In particular, it is our preliminary assessment that such software engineering integration efforts should begin by articulating the common topological aspects as a basis for the development of a supporting topological subsystem. Such a subsystem would be widely portable and would serve as the basis for new topological design tools, much in the way existing graphical subsystems support the development of a wide range of graphics applications.

In summary, we propose that the cross-disciplinary integration of topology and design theory may afford significant opportunities for formalization of engineering design theory.

Acknowledgements

The authors gratefully acknowledge the anonymous reviewers for their thorough, substantive reviews and their many constructive comments. The authors also thank S. M. Dorney, D. A. Field, D. A. Lear, M. D. Rice, V. Shapiro, N. F. Stewart, and K. J. Weiler for their helpful comments. This work of T. J. Peters has been partially supported by the National Science Foundation under Grant Number MII-9308346 and by ARPA under Grant F29601-94-K-0043. The views expressed herein are of this author, not the National Science Foundation or ARPA. Similarly, this work of D. W. Rosen has been partially supported by the National Science Foundation under Grant Numbers DMI-9414715 and DMI-9420405.

References

Alpha_1 (1990) User's Manual, Department of Computer Science, University of Utah.

ANSI Standard Y14.5M (1982) *Dimensioning and Tolerancing*, American Society of Mechanical Engineers, New York, NY.

Andersson, L-E., Dorney, S. M., Peters, T. J. and Stewart, N. F. (1994) Polyhedral perturbations that preserve topological form, *CT Technical Report CSE-TR-94-09*, University of Connecticut, Storrs. To appear in *Computer-Aided Geometric Design*.

Besl, P. (1994) SurfSeg V3.9 User/Reference Manual, GM R&D Center, Warren, MI.

Bøhn, J. H. and Wozny, M. J. (1993) A topology-based approach for shell-closure, In *IFIP Transactions, Geometric Modeling for Product Realization*, P. R. Wilson, M. J. Wozny, and M. J. Pratt (eds) August, North-Holland, Amsterdam, 297–320.

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

Boyer, M. and Stewart, N. F. (1992) Imperfect form tolerancing on manifold objects: A metric approach, *International Journal of Robotics Research*, 11(5), pp. 482–490.

Cunningham, J. J. and Dixon, J. R. (1988) Designing with features: The origin of features , In *Proc. ASME Computers in Engineering Conference*, San Fransisco, pp. 237–244, August 1–4.

Desaulniers, H. and Stewart, N. F. (1992) An extension of manifold boundary representations to the R-sets, *ACM Transactions on Graphics*, 11(1), 40–60, January.

Dorney, S. M., (1994) Maintaining equivalent topological form in tolerance modeling, Doctoral Dissertation, Department of Computer Science and Engineering, University of Connecticut, Storrs, CT.

Gadh, R., Gursoz, E. L., Hall, M. A., Prinz, F. B. and Sudhalkar, A. (1991) Feature abstraction in a knowledge-based critique of designs, *Manufacturing Review*, 4(2), 115–125.

Gursoz, E. L., Choi, Y. and Prinz, F. B. (1988) Vertex-based representations of non-manifold boundaries, In *Geometric Modeling for Product Engineering*, IFIP WG5.2, Rensselaerville, NY, M. J. Wozny, J. U. Turner, and K. Preiss (eds), North-Holland, Amsterdam, pp. 107–130.

Hocking, J. G. and Young, G. S. (1961). *Topology*, Addison-Wesley, Reading, MA.

IFIP (1992) Special discussion session, *Non-manifold environments: issues and futures*, Fourth IFIP WG 5.2 Workshop on Geometric Modeling in Computer-Aided Design, Sept. 27–Oct. 1, Rensselaerville, NY.

Juster, N. P. (1992) Modeling and representation of dimensions and tolerances: A survey, *Computer-Aided Design*, 24(1), 3–17.

Kiriyama, T., Tomiyama, T. and Yoshikawa, H. (1991) The use of qualitative physics for integrated design object modeling, In *Proceedings ASME Design Theory and Methodology Conference*, pp. 53–60.

Lear, D. A. (1992) Practical applications of algebraic topology to geometric design, working paper, Workshop Notes, Tab 3, Fourth IFIP WG 5.2 Workshop on Geometry Modeling in Computer-Aided Design, Sept. 27–Oct. 1, 1992, Rensselaerville, NY.

Mäntylä M. (1988) *An Introduction to Solid Modeling*, Computer Science Press.

Massey, W. S. (1967) *Algebraic Topology: An Introduction*, Harcourt Brace Jovanovich, New York.

Munkres, J. R. (1975) *Topology, a First Course*, Prentice Hall, Inc., Englewood Cliffs, New Jersey.

Palmer, R.S. and Shapiro, V. (1993) Chain models of physical behavior for engineering analysis and design, *Research in Engineering Design*, 5(3/4), 161–184.

Palmer, R. S. (1995) Using the *Chains* algebraic-topological language for finite element analysis, *Computed Aided Geometric Design*, to appear.

Peters, T. J., Demurjian, S. A., Ting, T. C. and Glovin, S. (1994*a*) Feature-based modeling by object-oriented design with propagation. In *Proceedings of 1994 International Conference on Data and Knowledge Systems for Manufacturing and Engineering*, Hong Kong, May, pp. 101–110.

Peters, T. J., Rosen, D. W. and Dorney, S. M. (1994*b*) The Diversity of Topological Applications within Computer-aided Design, Papers on General Topology and Applications, In *Proceedings of the Eighth Summer Conference at Queens College, Annals of the New York Academy of Sciences*, S. Andima *et al.* (eds.), 728, 198–209.

Peters, T. J., Rosen, D. W. and Shapiro, V. (1994*c*) A topological model of limitations in design for manufacturing, *Research in Engineering Design*, 6(4), 223–233.

Poli, C., Escudero, J. and Fernandez, F. (1988) How part design affects injection molding tool costs, *Machine Design*, Nov. 24.
Reddy, G. and Cagan, J. (1995) An improved shape annealing algorithm for truss topology generation, *ASME Journal of Mechanical Design*, 117(2A), 315–321.
Requicha, A. A. G. (1980) Representations for rigid solids: Theory, methods, and systems, *ACM Computing Surveys*, 12(4), 437–464.
Requicha, A. A. G. (1983) Toward a theory of geometric tolerancing, *International Journal of Robotics Research*, 2(4), 45–60.
Requicha, A. A. G. and Rossignac, J. R. (1992) Solid modeling and beyond, *IEEE Computer Graphics and Applications*, September, 31–44.
Requicha, A. A. G. (1993) Mathematical definition of tolerance specifications, *Manufacturing Review*, 6(4), 269–274.
Requicha, A. A. G. and Voelcker, H. (1982) Solid modeling: A historical summary and contemporary assessment, *IEEE Computer Graphics and Applications*, March, 2, 9–24.
Rosen, D. W. (1992) A feature-based representation to support the design process and the manufacturability evaluation of mechanical components, Doctoral Dissertation, University of Massachusetts, May.
Rosen, D. W., Dixon, J. R., Poli, C. and Dong, X. (1992) Features and algorithms for tooling cost evaluation in injection molding and die casting, *Proc. ASME Computers in Engineering Conference*, Vol. 1, pp. 45–52.
Rosen, D. W. and Peters, T. J. (1992) Topological properties that model feature-based representation conversions within concurrent engineering, *Research in Engineering Design*, 4(4), 147–158.
Shah, J. J. (1991) Assessment of features technology, *Computer-Aided Design*, 23(5), 331–343.
Shapiro, V. (1991) Representations of semi-algebraic sets in finite algebras generated by space decompositions, Doctoral Dissertation, Sibley School of Mechanical and Aerospace Engineering, Cornell University.
Shapiro, V. and Voelcker, H. (1989) On the role of geometry in mechanical design, *Research in Engineering Design*, 1(1), 69–73.
Srinivasan, V. (1993) The role of sweeps in tolerancing semantics, *Manufacturing Review*, 6(4), 274–281.
Stewart, N. F. (1993) A sufficient condition for correct topological form in tolerance specification, *Computer-Aided Design*, January, 25(1), 39–48.
Sudhalkar, A., Gursoz, L. and Prinz, F. (1993) Continuous skeletons of discrete objects. In *Proceedings 2nd ACM/IEEE Symposium on Solid Modeling and Applications*, Montreal, pp. 85–94.
Taura, T. and Yoshikawa, H. (1991) Metric space for intelligent CAD. In *Proceedings of IFIP WG 5.2*, Columbus, OH, September 30–October 3, 1991, D. C. Brown, M. B. Waldron, H. Yoshikawa (eds.), North-Holland, Amsterdam.
Tilove, R. B. and Requicha, A. A. G. (1980) Closure of Boolean operations on geometric entities, *Computer-Aided Design*, 12(5), 219–220.
Tomiyama, T., Kiriyama, T., Takeda, H., Xue, D. and Yoshikawa, H. (1989) Metamodel: A key to intelligent CAD systems, *Research in Engineering Design*, 1 (1), 19–34.
Tomiyama, T. and Yoshikawa, H. (1985) Extended general design theory. In *Proceedings of IFIP WG 5.2*. Tokyo, North-Holland, Amsterdam, pp. 95–130.
Voelcker, H. (1990) The "Metrology Crisis"—Status report 1, Working paper for The Manufacturing Studies Board, February.
Voelcker, H. (1991) The "Metrology Crisis"—Status report 2, Working paper for The Manufacturing Studies Board, February.
Voelcker, H. (1993) The "Metrology Crisis"—Status report 4, Working paper for The Manufacturing Studies Board, February.
Ward, A. C. (1990) A recursive process for managing the design process. In *Proceedings ASME Design Theory and Methodology Conference*, pp. 47–52.
Ward, A. C. (1992) Some language-based approaches to concurrent engineering, *International Journal of Systems Automation: Research and Applications*, 2(4), 335–351.
Weiler, K. J. (1986) Topological structures for geometric modeling, Doctoral Dissertation, Rensselaer Polytechnic Institute.
Weiler, K.J. (1988) The radial edge structure: A topological representation for non-manifold geometric boundary modeling. In *IFIP WG5.2 Working Conference on Geometric Modeling for CAD Applications*. Rensselaerville, NY, May 12–14, 1986, M. J. Wozny, H. McGlaughlin, and J. Encarnacao (eds.), North-Holland, Amsterdam, pp. 3–36.
Weiler, K. J. (1992) Personal communication, December 23.
Willard, S. (1970) *General Topology*, Addison-Wesley, Reading, MA.
Yoshikawa, H. (1981) General design theory and a CAD system. In *Man–Machine Communications in CAD/CAM*, Tokyo, October 2–4, 1080, Proceedings of IFIP WG 5.2, North-Holland, Amsterdam, pp. 35–38.
Zamanian, M. K., Fenves, S. J. and Gursoz, E. L. (1992) Representing spatial abstractions of constructed facilities, *Building and Environment*, 27(2), 221–230.

Appendix: Topology Fundamentals

These elementary topological concepts are critical to the preceding presentation of a theoretical basis for design. Definitions A.1 to A.10 are taken from Munkres (1975).

Definition A.1 A *topology* on a set X is a collection $\mathscr{I}$ of X having the following properties:

- Both ϕ and X are elements of $\mathscr{I}$.
- The union of the elements of any subcollection of $\mathscr{I}$ is an element of $\mathscr{I}$.
- The intersection of the elements of any finite subcollection of $\mathscr{I}$ is an element of $\mathscr{I}$.

As an example, consider the example topologies on a set of three elements shown in Fig. A.1.a. The situation depicted in Fig. A.1.b fails to be a topology because it does not satisfy the union criterion in the preceding definition of a topology.

The set X, with its associated topology $\mathscr{I}$, is called a *topological space*. When confusion is unlikely, the

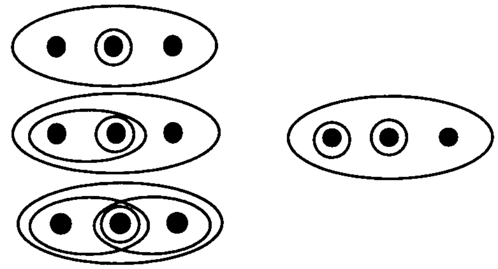


Fig. A.1. Examples of valid and invalid topologies on a set of three elements.

"topological" adjective is often suppressed, but is understood to be implied by more specialized modifiers. Thus, design and manufacturing spaces discussed in this paper, as well as general metric (see Definition A.7) spaces, are all examples of topological spaces.

Definition A.2 If X is a set with an associated topology $\mathscr{I}$, then each element of $\mathscr{I}$ is called an *open* subset of X. For any open subset U of X, its complement $X - U$ is said to be *closed*. If $x \in U \in \mathscr{I}$, then U is said to be a *neighborhood* of x.

A trivially defined, yet important, example of a topology on a set is that which considers each subset to be open. This condition necessarily implies that each subset is also closed. This topology is known as the *discrete topology*.

Many times it is impractical or difficult to specify a topology by describing each of its open sets. It is often more practical to specify a subcollection of subsets that can be used to generate the topology. The requirements for two such subcollections are given, in Definitions A.3 and A.5, below.

Definition A.3 If X is a set, a *basis* for a topology on X is a collection $\mathscr{B}$ of subsets of X (called *basis elements*) such that:

- For each $x \in X$, there is at least one basis element B containing x.
- If x belongs to the intersection of two basis elements B_1 and B_2 then there is a basis element B_3 containing x such that $B_3 \subseteq B_1 \cap B_2$.

In Munkres (1975), the symbol $\subset$ is used to indicate a *subset* relation, while we prefer the symbol $\subseteq$ for *subset* and $\subset$ for *proper subset*.

Lemma A.4 Let X be a set; let $\mathscr{B}$ be a basis for a topology $\mathscr{I}$ on X. Then $\mathscr{I}$ equals the collection of all unions of elements of $\mathscr{B}$.

Definition A.5 If X is a set, a *subbasis* $\mathscr{S}$ for a topology on X is a collection of subsets of X whose union equals X. The *topology generated by the subbasis* $\mathscr{S}$ is defined to be the collection $\mathscr{I}$ of all unions of finite intersections of elements of $\mathscr{S}$.

The subsequent development relies heavily upon understanding the standard topology on the real numbers. The basis elements generating this topology are all open intervals of real numbers of the form

$$(a, b) = \{x \mid a < x < b\}$$

Given any set, it may be desirable to measure the difference between two elements of that set (for instance, to measure the difference between two different designs). A metric is the standard mathematical formalism for quantifying such differences. In general, defining a metric on a set induces a topology on that set, where neighborhoods in the topology are given by ε-balls, as defined below. In the following, the notation **R** will represent the set of real numbers.

Definition A.6 A *metric* on a set X is a function $d: X \times X \to \mathbf{R}$ with the following properties:

1. $d(x, y) \geq 0$ for all $x, y \in X$; equality holds if and only if $x = y$;
2. $d(x, y) = d(y, x)$ for all $x, y \in X$;
3. $d(x, y) + d(y, z) \geq d(x, z)$ for all $x, y, z \in X$ (triangle inequality).

The number $d(x, y)$ is called the *distance* between x and y in metric d. Given $\varepsilon > 0$, the set

$$B_d(x, \varepsilon) = \{y \mid d(x, y) < \varepsilon\}$$

is called the ε-ball centered at x (Munkres, 1975)

Definition A.7 If d is a metric on a set X then the collection of all *ε-balls* $B_d(x, \varepsilon)$ for $x \in X$ and $\varepsilon > 0$ is a basis for a topology on X, called the *metric topology* induced by d.

The following definitions and theorems concern the notion of topological connectedness. Connectedness properties of design and manufacturing spaces are initimately related to the continuity of the corresponding conversion functions.

Definition A.8 Let X and Y be topological spaces and let h be a function, $h: X \to Y$. Then h is a *homeomorphism* if and only if h is one-one, onto Y and both h and its inverse h^{-1} are continuous.

Definition A.9 A space X is *connected* if and only if the only subsets of X that are both open and closed in X are the empty set and X itself.

Definition A.10 A space X is *locally connected at* x if for every neighborhood U of x, there is a connected neighborhood V of x contained in U. If X is locally connected at each of its points, it is said simply to be *locally connected*.

The real line, with its usual topology is both connected and locally connected, but these properties need not co-exist. Furthermore, since the topology of higher dimensional real spaces (such as 2D, 3D, etc.) are constructed from the real line, these higher dimensional spaces are also both connected and locally connected. At the opposite extreme from connected spaces are discrete spaces. Note that in a discrete space, each singleton set is both open and closed. The fundamental relationship between continuity and connectedness is expressed, below, in Definition A.11 and Theorem A.12.

Definition A.11 Let X and Y be topological spaces and let h be a function h: $X \rightarrow Y$. Then h is *continuous* at $x_0 \in X$ if and only if for each neighborhood V of $h(x_0)$, there is a neighborhood U of $x_0 \in X$ such that $h(U) \subset V$. It is said that h is *continuous on* X if and only if h is continuous at each $x_0 \in X$.

Theorem A.12 A continuous functions maps a connected space onto a connected space.

Theorem A.12 is often paraphrased as "Continuous functions preserve connectedness". Note that it immediately follows that any function from a connected space onto a disconnected space cannot be continuous. In particular, a conversion function h from a connected design space D onto a disconnected manufacturing space M cannot be continuous. In fact, we shall argue that many traditional manufacturing spaces are indeed badly disconnected. However, some additional subtlety beyond direct application of Theorem A.12 is required, as many of these design spaces are merely locally connected, but not connected. Nonetheless, Theorem A.12 remains a fundamental tool in that discontuinity argument.

The following definition has been specialized to be directly applicable within the context of this paper.

Definition A.13 Let g be a function from the set X onto the topological space Y. Let $\mathscr{I}(Y)$ be the collection of open sets of Y. Then the collection $\{g^{-1}(U) \mid U \in \mathscr{I}(Y)\}$ can be used as a subbasis that generates a topology on X. Such a topology is called the *weak topology*.

It is trivial to observe that under this weak topology, g is necessarily continuous by construction. Although this weak topology guarantees the continuity of g it does so at the potential expense of X losing some topological properties that may have resulted from a more natural, intuitively pleasing topology on X. For instance, it may have been natural for X to have a connected, metric topology. However, the corresponding weak topology may not be connected or metric.

Definition A.14 appears in Willard (1970).

Definition A.14 A space X is *extremally disconnected* if and only if each point $x \in X$ has a collection of basis elements containing x such that each member of the collection is both open and closed.

In particular, any discrete topological space X is *extremally disconnected.*