On the Synthesis of Functionally Equivalent Mechanical Designs

Horea Ilieş
Ford Motor Company*
hilies@ford.com

Vadim Shapiro
University of Wisconsin-Madison†
vshapiro@engr.wisc.edu

Abstract

Conventional mechanical design focuses on a single completely specified nominal shape that is later toleranced to allow for variations in form. The corresponding design processes usually involve arbitrary decisions affecting the geometry and do not support systematic generation of alternative shapes satisfying identical or altered functionalities. This places a serious handicap on the design cycle of a product, since most new designs are obtained by modifying existing products to comply with new functional specifications.

Since the functionality of a part does not usually define all of its geometry, a more coherent approach would be to design classes of equivalent mechanical parts that satisfy a given functionality. We show here that, by replacing the completely specified geometry of the traditional approaches with partial geometry *and* functional specification, we can generate classes of mechanical parts that are equivalent, in the sense that all members of the class satisfy the same functional specifications.

Ability to define, compute, and represent classes of functionally equivalent parts will allow one to generate, compare and modify functionally equivalent designs, perhaps having dissimilar geometries. We identify three such equivalence classes for artifacts with parts moving in contact, which presume, at the very least, contact between parts, spatial containment during their relative motion, and external loads applied on the moving parts. We show that these classes of functionally equivalent parts are computable and may be represented unambiguously by maximal elements of each class.

Keywords: Equivalence classes, functional equivalence, shape synthesis, design space, higher pairs, contact, sweep, unsweep, moving parts.

Introduction

Mapping engineering function into a functional form is, by all accounts, the central activity of engineering design, and has been one of the drivers of economic and social development throughout human history. At the same time, mechanical design has remained conceptually unchanged in the sense that traditional approaches to mechanical design, still widely used today, fully rely on the creativity and engineering intuition of the designers and lead to design procedures that are intensely iterative.

Despite their well known benefits, these essentially "generate and test" scenarios are costly and time consuming during the product development process and their effectiveness is directly dependent on the available "design experience", as well as on the time and resources that are being allocated for product development. A comprehensive review of mechanical design literature is outside the scope of this paper. Broadly, the existing approaches to conceptual design focus on formulating a set of design rules to be satisfied (Suh 1990), on applying various search techniques to find a solution for the design problem or on interactive (and iterative) specialized design approaches, such as knowledge-based reasoning. While the solution search techniques, such as "path finding", "constraint satisfaction", "simulated annealing", "genetic algorithms" and various shape optimization techniques (Stahovich 2001; Lee, Ma, & Antonsson 2001), are becoming more popular, they typically lead to single designs with fully specified geometries. For parts moving in contact, the recently proposed methods lead to iterative analysis techniques that are primarily based on configuration space computations (Caine 1993; Joskowicz & Addanki 1988; Joskowicz & Sacks 1999).

Consider the automotive latch shown in Figure 1, which is part of a typical hood latch assembly of an automobile. Broadly, the functionality of a hood latch assembly is to engage and retain a (generally cylindrical) striker[1] *and* to prevent it from accidental release. During its downward motion, a vertically moving striker forces the secondary latch to rotate and latch it. While the striker continues to move vertically down under the influence of externally applied loads, the secondary latch is brought back to its original position by an attached reaction spring; the primary latch, not shown, closes down and holds the striker fixed. The release of the striker from the hood latch assembly is accomplished in two phases: first, manually applying external forces indirectly to the primary latch from the cabin, and then directly to the

*Please use e-mail as means of contact. Alternate e-mail address: ilies@sal-cnc.me.wisc.edu

†Spatial Automation Laboratory, Department of Mechanical Engineering, 1513 University Avenue, University of Wisconsin-Madison, 53706 USA.

[1] In this example the striker is attached to the hood of the automobile.

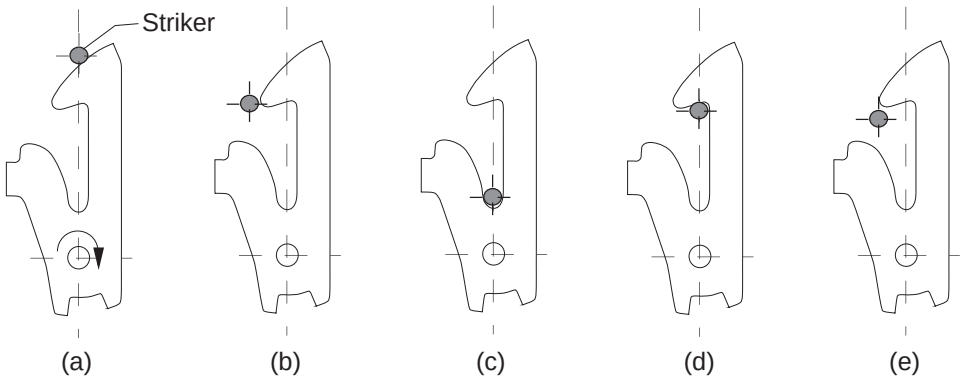


Figure 1: A secondary hood latch must engage and retain a vertically moving striker.

secondary latch – so that the secondary latch rotates and the hood can be lifted. A spring (not shown) attached to the secondary latch provides the reaction force needed to insure the return to its original position, but the latch must continue to function even if the spring breaks. The secondary latch must also prevent the upward motion of the striker (and therefore of the hood) when the primary latch fails to engage the striker. In addition, the secondary latch has to remain inside a specified containing set during its motion to avoid interference with neighboring parts.

We focus on the *contact function* of the latch, since engaging the striker and avoiding its accidental release is accomplished by the contact between the striker and the latch. Though the design of such a latch is also affected by other constraints, including strength, containment, and manufacturability, as discussed in (Ilieş & Shapiro 1996; 1999). At the very least, the contact function presumes that (1) the moving parts are positioned relative to each other so that their boundaries always touch each other; (2) each part is contained in its work space without interfering with other parts; and (3) the contact surfaces are able to support the externally applied loads on the contacting parts. Specifically, in the case of the latch shown in Figure 1, the latch comes in contact with the striker, it must not interfere with other components, and external forces are applied indirectly to the striker through the hood of the automobile.

We can use this common automotive latch to make few essential observations that are central to the rest of the paper. The functionality of the latch determines portions of functional geometry, and *all* functional surfaces are determined by some intended function of the latch. On the other hand, non-functional surfaces can be regarded as opportunities to improve the performance of the part. This suggests that there is a *class* of (infinitely many) functionally equivalent designs that satisfy the same mechanical function. Thus, there are infinitely many shapes that satisfy the specified containment constraints for a given relative motion. At the same time, there are infinitely many shapes (some of them unbounded) that can move in contact with the striker according to the known motion[2], but not all these shapes will also satisfy the containment constraints. Among the latch shapes that can move in contact with the striker and remain inside the containing workspace, some may not latch the striker for a given set of externally applied loads. This would eliminate them from the feasible designs and effectively reduce the design space. It is intuitively clear that each of the three constraints induces a class of functionally equivalent designs, and any selected part must lie in the intersection of all three classes that still contains infinitely many parts.

This latch illustrates an example of a part that is often redesigned to accommodate new car models and/or changing requirements. Since the geometry of the latch, in itself, does not contain any information about the functionality of the latch, or about the reasons that led the designer to choose this particular geometry, the redesign process often requires guesswork or a complete re-design of the part. This leads to costly "generate and test" design procedures that may not capture the original design intent, because either the process may not converge, or the geometry may be overconstrained, or we may simply not know whether a solution exists (i.e., the corresponding equivalence class is empty).

In contrast, the ability to define, compute, and represent functionally equivalent classes allows one to generate, compare and modify functionally equivalent designs, perhaps having dissimilar geometries. In this paper we identify three such equivalence classes for artifacts with parts moving in contact under applied loads that are also subjected to containment constraints, show that they may be computed and represented unambiguously by maximal elements of each class. We show that, by replacing the completely specified geometry of the traditional approaches with partial geometry *and* functional specification, we can generate classes of mechanical parts that are equivalent in the sense that all members of the class satisfy the same functional specifications. Our approach captures the design decisions by generating

[2]The simplest way to generate two such shapes is to add non-functional holes to a functional latch. Intuitively, one can modify also non-functional surfaces without changing the function of the part.

functional surfaces that do not embed arbitrary geometric restrictions, and identifies fully defined representative members of the equivalence classes that contain an essentially unlimited number of functional designs.

Functional Equivalence Classes

An equivalence class is a set of elements that are related to each other via an equivalence relation "$\sim$" that must be, by definition, reflexive ($a \sim a$), symmetric ($a \sim b$ implies $b \sim a$) and transitive ($a \sim b$ and $b \sim c$ implies $a \sim c$). Equivalence classes become particularly important in design when one can define equivalent relation(s) that capture the functionality of a given part. Thus, any member of the equivalence class will be functionally equivalent with any other member of the same class. At the very least, such an equivalence class must have computable representative members, and well defined test for membership in the class. Below we discuss the three equivalence classes for the moving parts serving the contact function exemplified by the automotive latch above.

Containment Equivalence

Every moving part A needs to remain inside a given containing space E to prevent interference with other parts during its operation. In this paper a motion M is a one parameter family of transformations $M(t)$ from the normalized time domain $t \in [0, 1]$ to the space of geometric configurations $\mathcal{C}$ (Ilieş & Shapiro 1996). Then the containment constraint can be formally expressed as

$$A^{M(t)} \subset E, \ \forall t \in [0, 1] \tag{1}$$

where $A^{M(t)}$ denotes set A at instance t during its motion M, and the interval in which t lies has been normalized.

All sets that satisfy the condition (1) for a given containing set E and motion M are equivalent because they all serve the same containment function. We will refer to such equivalence class of containment as *containment equivalence* for short.

The above definition is not very useful for design purposes because it does not suggest a direct method for deciding whether two shapes A_1 and A_2 belong to the same containment equivalence class, or how a shape may be modified without violating the containment constraint. The problem is solved by identifying the unique *maximal* shape in the containment equivalence class. Formally, it is defined by the `unsweep` operation, which returns the largest moving set of points remaining inside a given containing set, and is basically a material removal operation (Ilieş & Shapiro 1999). As shown in Figure 2, the largest "latch" remaining inside the stationary containing set E during a known motion M is $\texttt{unsweep}(E, M)$ and contains *all* points of E that would remain inside E during M. This maximal set $\texttt{unsweep}(E, M)$ defines an equivalence class in the sense that any of its subsets will satisfy the same containment constraints. In practice, it can be computed based on the two equivalent definitions [3] of `unsweep`, one in terms of a trajectory test, and the other in terms of an infinite intersection of the moving set:

$$\begin{aligned} \texttt{unsweep}(E, M) &= \textstyle\bigcap_{q \in \hat{M}} E^q \\ \texttt{unsweep}(E, M) &= \{x \mid T_x \subset E\} \end{aligned} \tag{2}$$

where T_x is the trajectory of a moving point $x \in E$.

Positional Equivalence

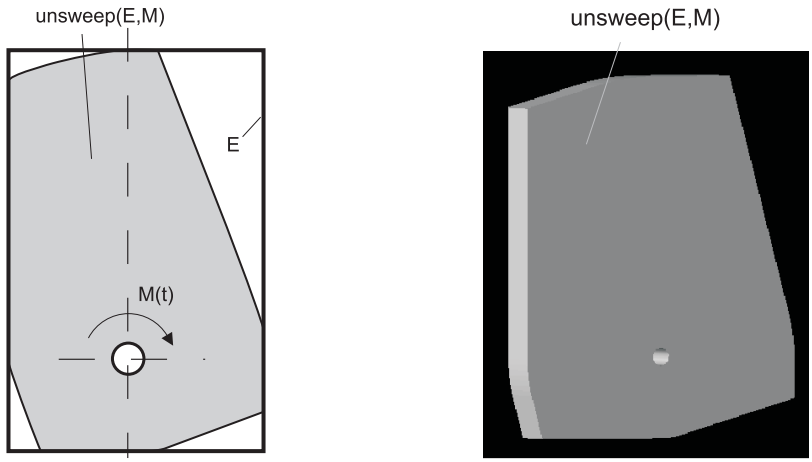
Informally, moving parts that do not interfere may or may not serve the desired contact function, depending on whether their respective boundaries are touching each other at all times during the motion. On the other hand, all pairs of shapes that can execute the same motion while touching each other may be deemed *positionally equivalent.* We introduced *conjugate triplets* in (Ilieş 2000; Ilieş & Shapiro 2002) as a means of representing and manipulating classes of positionally equivalent parts. A conjugate triplet $\Psi = \langle A, B, M \rangle$ consists of two shapes A and B moving relative to each other according to a motion M while touching each other. As illustrated in Figure 3(a-c), a triplet may not exist for given shapes and motion, but if one does exist, then there are infinitely many such conjugate shapes for the same motion. This assertion naturally leads to the notion of positionally equivalent triplets, which share the same set of configurations for their motions and satisfy the inclusion relationship between their respective contact boundaries (see (Ilieş & Shapiro 2002) for formal definitions). Once again, rather than comparing triplets directly, we may seek a unique maximal triplet capable of representing the corresponding equivalence class. Starting with a given triplet $\Psi = \langle A, B, M \rangle$, let us fix motion M and part A, but let B 'grow' while its boundaries remain in contact with A at all times. This process may produce additional contacts between A and B, but can not eliminate the contacts that are already present. Thus Ψ is guaranteed to be in the positional equivalence class of the grown triplet. The growing process stops when the largest possible object B^A_{max} cannot be grown without violating the contact constraints. But now we can also grow object A to obtain the new set A^A_{max} that remains in contact with B^A_{max} for the same motion M. Formally,

$$\begin{aligned} B^A_{max} &= k[(\texttt{sweep}(A, M))^c] \\ A^A_{max} &= k[(\texttt{unsweep}((B^A_{max})^c, M))] \end{aligned} \tag{3}$$

where X^c denotes the standard set complement of a set X, and kX is the closure of X. The properties of `sweep` and `unsweep` operations imply that for a given Ψ, both B^A_{max} and A^A_{max} are well defined, and in this sense Ψ^A is *unique*. Note that $A \subset A^A_{max}$, based on the properties of the `unsweep` operation discussed in (Ilieş & Shapiro 1999). In other words, A^A_{max} is the largest A that would generate the largest B during the given M.

Figures 4(a-c) show the largest latch shape B^A_{max} that moves in contact with the striker A according to motion M.

[3] By using the first definition, one can prove that `unsweep` is dual to the general `sweep` operation. The second equation leads to PMC tests for both dual operations that classify given points as 'in', 'on', or 'out' of the corresponding sets.



(a) (b)

Figure 2: The largest shape (set of points) remaining inside a containing set E during a known motion $M(t)$ is given by $\texttt{unsweep}(E, M)$.

It is obvious that the set shown in Figures 4(a-c) is not related from a geometrical point of view to the set shown in Figure 2. In other words, the positional equivalence (which encodes the contact constraints) is *not* a special case of containment equivalence (which encodes the containment constraints). The largest striker A^A_{max} (not shown) follows from equations (3).

Because we have a choice of which object to grow first, the above process is clearly order-dependent and asymmetric with respect to A and B. Therefore there exists another maximal conjugate triplet $\Psi^B = \langle B^B_{max}, A^B_{max}, \hat{M} \rangle$ defined as in (3) with A and B interchanged, with the maximal set A^B_{max} that will come in contact with B while B moves relative to A^B_{max} according to $\hat{M}$. Thus, in principle, we could fix the shape of the latch and design the largest possible striker that works with this latch, even if this does not appear practical for this particular application.

To summarize, any given triplet Ψ naturally belongs to two unique and distinct equivalence classes that are represented unambiguously by *two* maximal triplets. These triplets are made up from the largest (and hence unique) shapes that maintain the specified motion and contact. If the initial object A is given, then the function of the triplet is to maintain a moving contact with the given A, and we focus on the properties of the maximal triplet Ψ^A. Changes to A are permitted *only* to the extent that they do not affect the shape of the largest conjugate shape B, because this would effectively lead to a new design problem. By symmetry, the maximal triplet Ψ^B should be used when designing the shape of A to move in contact with a partially known object B.

Static Equivalence

The positional equivalence class discussed above captures the contact constraints imposed on a given part by identifying the boundary of a shape B which contains *all* points that will come in contact with a given object A during a prescribed motion M. But for any given set of applied external loads, only some of these contact points will serve the contact function by carrying loads and transmitting the contact forces. This implies that those points that are not carrying loads can be eliminated without violating the contact function.

For example, Figure 5 shows a cylinder rolling on a planar surface under an applied external force $\mathbf{F}$ and set B_1 contains all contact points of the planar surface that carry the load $\mathbf{F}$. In general, not all these boundary points B_1 are required by a given $\mathbf{F}$, and Figure 5(b) shows a subset B_2 of B_1 that may be sufficient to carry the applied load. It follows that the two sets B_1 and B_2 perform the same function and are functionally equivalent in this sense.

Informally, the *static equivalence* class identifies all shapes that share boundary points required by the contact constraints and the applied loads. Our definition assumes quasi-static relative motion of the parts – so that inertia effects are negligible, and relies on a classification of all the boundary points of the maximal triplets into *load-bearing*, (the contact points that carry the applied loads) and *load-free* boundary points. The maximal triplets are a logical starting point because, by definition, the boundaries of the maximal parts contain *all* points where load-bearing contact may take place.

Figure 6 shows the maximal triplet corresponding to a cylindrical follower A and known relative motion M, while an external force $\mathbf{F}$ is applied on A so that $\mathbf{F}$ has a component collinear and opposite to the surface normal at the contact point throughout the relative motion. More generally, at a given contact point between A and B, there can be a contact force that is either positive (one object "supports" the other) or zero. We do *not* assume frictionless contact, but observe that when the normal contact force at a given contact point is zero, the friction force at that point is also zero.

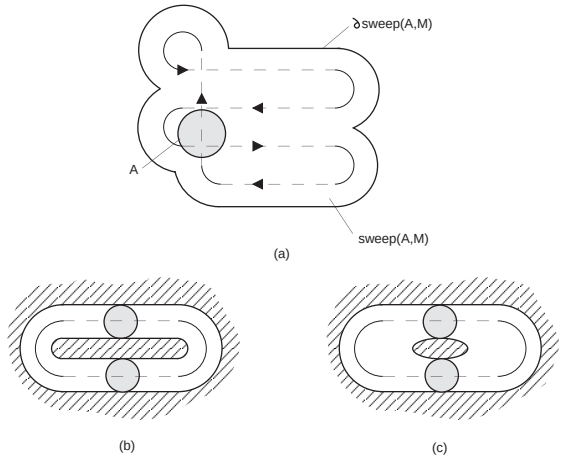


Figure 3: A conjugate triplet does not always exist for given object and motion (a). If one conjugate triplet exists (b), there are infinitely many conjugate triplets one of which is shown in (c).

The contact points at which the objects "support" each other are *load-bearing*, while the other contact points are *load-free*. The set of all load-bearing points form the load-bearing surfaces, while the union of all the load-free points are the load-free surfaces. Formal definitions of *load-bearing* and *load-free* points and surfaces can be found in (Ilieş 2000).

For a quasi-static motion, determining whether a given contact point P is load-bearing requires three pieces of information (Ilieş 2000): the directions of the external loads on the two contacting objects (which are generally known), the coordinates of P and information about the surface normal at P. Since we have complete geometric information about the objects, the contact points and their normals are easily computed. For the example shown in Figure 6(a), all contact points of the outer profile of B are load-bearing, while all the contact points of the inner profile are load-free. Changing the direction of the externally applied loads may change the spatial distribution of the load-bearing points. For example, all load-bearing points are part of the boundary of the inner profile in Figure 6(b). More interesting may be a situation where the load-bearing points are shared by both the inner and outer profiles of B for certain loading conditions as illustrated in Figure 6(c). Note that the contact function of the mechanism requires load-bearing points for every relative configuration between A and B.

This classification of spatial points into load-bearing and load-free implies that the static equivalence class is a subclass of the positional equivalence class discussed in the previous Section. Only those members of the positional equivalence class that contain load-bearing contact points at every configuration of the relative motion will be members of the static equivalence class. So in a sense, the maximal triplets provide the conceptual transition from complete geometry to geometric function, and *now* to a non-geometric loadbearing function.

Needless to say that the mere existence of the load-bearing contact points will not guarantee the proper functionality of a given triplet which is subjected to some external loads along the known directions. A more complete validation must include strength and motion analysis which is impossible without knowing the actual load magnitudes and profiles. Nevertheless, the static equivalence class is important in its own right, because it contains all shapes that the satisfy the necessary conditions of the contact function, without over-constraining the actual shape of the parts. In particular, the load-free points and surfaces can be changed freely, without violating the contact constraints.

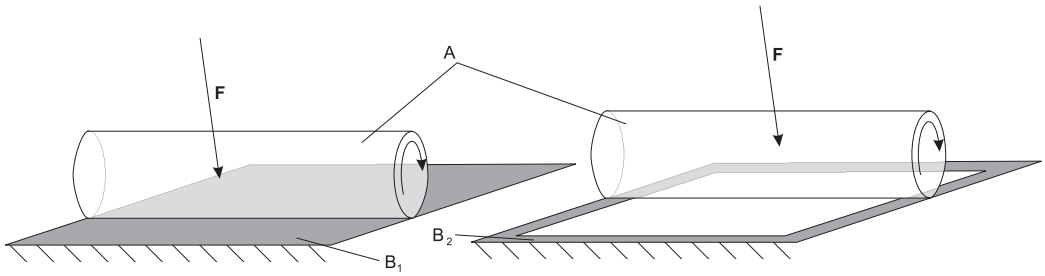
Synthesis Through Equivalence Classes

To demonstrate the advantages of design in terms of equivalence classes, consider again the automotive secondary latch illustrated in Figure 1 may be designed in terms of the identified equivalence classes. In the initial stages of the design of such a latch, the designer does not have an analytical representation for the relative motion M, but rather has a feeling for how the relative motion between the striker and the latch may look like. We assume that such a partial understanding of the relative motion is properly expressed by an incomplete description of the motion through a set of relative configurations of the striker and the latch to be designed. Such a set of relative configurations were interpolated here using the algorithm described in (Horsch & Jüttler 1998), which resulted in a continuous description of the motion of the striker relative to the latch. Different design cases may require different interpolation algorithms producing motions with dissimilar properties: in some cases the smoothness and continuity of the resulting motion can be essential, while in others, the simplicity of the resulting shape may be more important (Ilieş 2000).

The corresponding maximal triplet Ψ_A represents the class of all *positionally* equivalent latches B^A_{max} that move in contact with striker A. Once the motion M is known, it is

(a) (b) (c)

Figure 4: The largest connected latch, which includes all other positionally equivalent shapes moving in contact with the given striker according to a known motion. It corresponds to set B^A_{max} in equation (3).



(a) (b)

Figure 5: A cylinder rolling on a plane under the influence of an externally applied force. Set B_1 in (a) shows all contact points between the cylinder and the plane. Set B_2 shows a subset of B_1 that *may* be sufficient to carry the applied force.

straightforward to compute the maximal shape B^A_{max}. If we require the latch to be a single part, it should be a connected set, and we must be able to maintain the contact with the given striker A using only one of the connected components of B^A_{max}.

By identifying all the load-bearing points of B^A_{max} we generate the class of statically equivalent latches which will contain the functional surfaces of any latch satisfying the same contact function. A vertical external force is exerted upon the striker, which first moves the striker down and then up. The statically equivalent connected shape represents the largest functional shape satisfying the given contact constraints and is shown in Figures 7(a-c), where the functional surfaces are shown in lighter color. It contains all other *statically equivalent* shapes satisfying the specified contact function. One can also test whether new geometries of the latch satisfy the contact function by performing regularized set operations and simple geometric computations to confirm the existence of load-bearing points for every configuration of the striker's motion relative to the latch.

The (maximal) solution satisfying the containment constraints is obtained by computing the representative member of the *containment equivalence* class (Figure 7(d)) which is guaranteed to contain all shapes satisfying the same containment constraints. Clearly, any functioning latch must be a member of the containment equivalence class.

Any final latch subjected to the same external loads must form a triplet with the striker and their relative motion. This triplet must be *statically equivalent* with the maximal triplet shown in Figures 7(a-c) and be part of the *containment equivalence* class represented by the largest shape in Figure 7(d). The largest such latch satisfying both contact and containment constraints is shown in Figure 7(e) and is obtained by intersecting the shapes shown in Figures 2 and 7(a-c). Once the result of the intersection is known, one must confirm that the latch has load-bearing points at every configuration of the relative motion between the striker and latch, otherwise a solution that satisfies all the constraints imposed does not exist.

Refining the geometry of this largest latch must *also* take

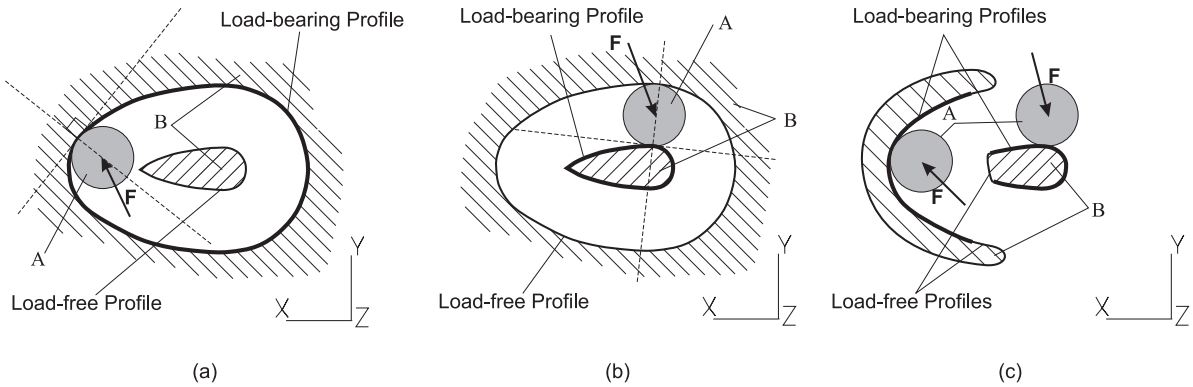


Figure 6: A cylindrical follower A maintains contact with B while moving under the influence of externally applied loads. For loading condition shown in (a) only the points on the outer profile of B are carrying the loads, but different loading conditions may redistribute these load-bearing points as shown in (b) and (c).

into account the strength of the latch, the manufacturing constraints and so on, but the functional latch must be part of the same static and containment equivalence classes. The non-functional surfaces of the latch can be changed without violating the contact function to accommodate additional constraints, or shape modifications that result from shape optimization algorithms.

Closure

Functional equivalence classes can play an essential role in design because they capture the design decisions without imposing arbitrary restrictions on the geometry of the parts. The traditional full geometric specification is replaced by partial geometry and functional specification. It should be clear that functional equivalence cannot be defined generally and uniquely, because this would also imply complete and *unique* characterization of the corresponding mechanical function.

This paper introduces three functional equivalence classes that capture the shape constraints induced by containment and contact constraints under quasi-static conditions. We showed that each of these three classes can be unambiguously represented by the maximal members that contain *all* other members in the same class that perform the same function. We have demonstrated that even these few equivalence classes of mechanical parts establish a new framework that is convenient – both conceptually and computationally – to the shape synthesis of moving parts. Within this framework one has direct access to the maximal elements of the equivalence class which contains all other functional designs, so that design space explorations can be performed systematically and efficiently. One can also test whether two designs, possibly having dissimilar geometries, are functionally equivalent (or belong to the same functional equivalence class) by performing relatively simple geometric computations.

An important and fruitful direction for future research is establishing other functional equivalence classes that may be derived and represented computationally. For example, assuming that the motion is quasi-static allowed us to neglect the inertia effects. Taking the kinetic phenomena into account would imply that the functional surfaces would not only depend on the geometry and motion configurations, but also on the first and second order derivatives of the motion as inertia is expected to become an important player.

Acknowledgments

This work was supported in part by the National Science Foundation grants DMI-0115133, CCR-0112758 and by the National Institute of Standards grant 60NANB2D0126.

References

Caine, M. 1993. *The Design of Shape From Motion Constraints*. Ph.D. Dissertation, Massachusetts Institute of Technology.

Horsch, T., and Jüttler, B. 1998. Cartesian spline interpolation for industrial robots. *Computer Aided Design* 30(3):207–224.

Ilieş, H., and Shapiro, V. 1996. An approach to systematic part design. In *Proceedings of the 5th IFIP WG5.2 Workshop on Geometric Modeling in CAD*, 383–392.

Ilieş, H., and Shapiro, V. 1999. The dual of sweep. *Computer Aided Design* 31(3):185–201.

Ilieş, H., and Shapiro, V. 2002. A class of forms from function: the case of parts moving in contact. *Research in Engineering Design* (13):157–166. A preliminary version of this paper appeared in the *Proceedings of the ASME 2001 DETC*, 13th *International Design Theory and Methodology Conference*.

Ilieş, H. T. 2000. *On Shaping Moving Mechanical Parts*. Ph.D. Dissertation, University of Wisconsin, Madison.

Joskowicz, L., and Addanki, S. 1988. Innovative shape design: A configuration space approach. Technical Report 356, Courant Institute of Mathematical Sciences, New York University.

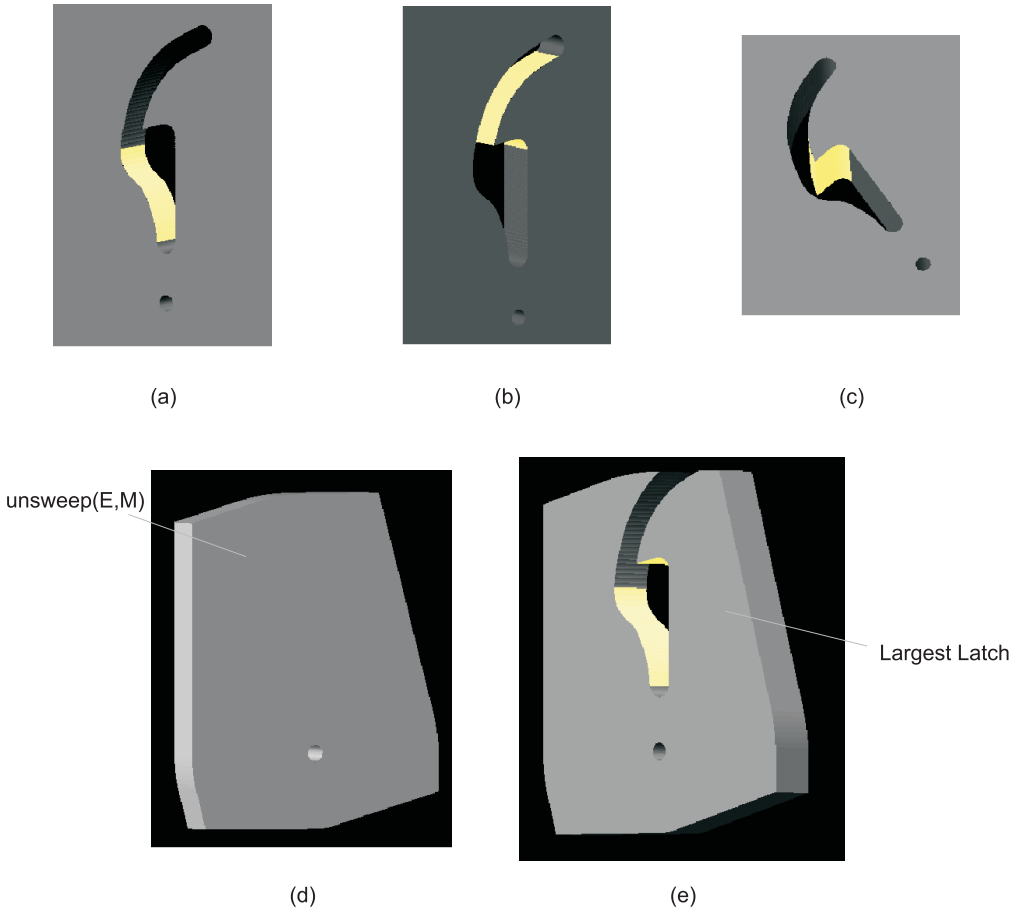


Figure 7: Figures (a-c) show the largest connected shape, which includes all other statically equivalent shapes satisfying the given contact function. The load-bearing boundary points are shown in lighter color. Figure (d) illustrates the largest shape satisfying the containment constraints. Figure (e) displays the largest latch satisfying both contact and containment constraints obtained through an intersection of the two corresponding maximal shapes.

Joskowicz, L., and Sacks, E. 1999. Computer-aided mechanical design using configuration spaces. *Computing in Science and Engineering* 1(6):14–21.

Lee, C.-Y.; Ma, L.; and Antonsson, E. 2001. Evolutionary and adaptive synthesis methods. In Antonsson, E. K., and Cagan, J., eds., *Formal Engineering Design Synthesis*. Cambridge University Press.

Stahovich, T. 2001. Artificial intelligence for design. In Antonsson, E. K., and Cagan, J., eds., *Formal Engineering Design Synthesis*. Cambridge University Press.

Suh, N. P. 1990. *The Principles of Design*. Oxford Series on Advanced Manufacturing.