Line Accessibility of Free Form Surfaces

Submission number: 80

Abstract

Given a three dimensional object $B \subset \mathbb{R}^3$, the visual hull of B (a concept introduced by A. Laurentini in 1992) is the smallest set $V_B \subset \mathbb{R}^3$ containing B with the following property: for each point p on the boundary of V_B there exists a direction from which p is on a silhouette of V_B. The notion of the visual hull has applications in computer graphics (visibility and silhouette based algorithms), manufacturing (accessibility, wire EDM and hot wire cutting), and more.

We present a tractable algorithm for computing a decomposition of a C^1 smooth, closed parametric surface $S \subset \mathbb{R}^3$, positioned in an arrangement with other occluding surfaces, into those regions that belong to the visual hull and those that do not (later referred to as the "line-accessible" or "line-inaccessible" regions). For efficiency, our method introduces two early domain pruning criteria, using a recursive subdivision of the parameter domain, along with a tangent plane bounding cone: one criterion detects regions that are guaranteed to belong to the visual hull, while the other detects regions that cannot belong to it (both use sufficient and not necessary conditions). Only to those sub-domains that remain after this significant domain reduction, an algebraic (subdivision based) solver is applied, to solve two sets of algebraic equations, already presented by A. Laurentini in 1999, gaining one to two orders of magnitude improvement in the computation times. We provide a derivation of these equations via a different approach, compared to A. Laurentini in 1999, using the model we refer to as the dynamic SSI solution of the surface and the moving tangent plane. Concrete computational examples are provided as well.

1 Introduction

From a variety of application domains, we choose to motivate the line-accessibility problem from the manufacturing field: assume we are required to manufacture a model, shaped as close as possible (with respect to the $\subseteq$ relation) to a closed surface $S \subset \mathbb{R}^3$, using hot wire or wire EDM techniques[1] on some initial stock material: clearly, not all possible regions of S can be produced exactly (even if we allow infinitely many wire maneuvers): some points might be inaccessible to the wire. This leads to the notion of line-accessibility, defined next. From now on we assume surface S is a regular parametric two-manifold, at least C^1 smooth. Further, we assume S is a closed surface (i.e. compact with no boundary). Denote by $T_P S$ the plane tangent to S at a point P. Then,

Definition 1. *A point $P \in S$ is called* line-accessible (LA) *if there exists a tangent direction $v \in T_P S$ such that the entire line through P with direction v has no common points with the interior of the volume bounded by S. The direction v is then called an* accessibility direction *or an* accessibility line. *If there's no such direction, P is called* line-inaccessible (LI). *When v has an argument angle θ with respect to some (orthonormal) basis of $T_P S$, the tangent line shall be denoted by l_θ, and θ is called an* accessibility angle.

[1] See, for example: http://en.wikipedia.org/wiki/Electrical_discharge_machining or http://en.wikipedia.org/wiki/Hot-wire_foam_cutter.

The connection of the visual hull concept to the line-accessibility problem is clear from Definition 1, as the existence of a direction from which a point can be viewed on a silhouette is equivalently stated as the line-accessibility of the point to the wire. The wire, in this context, is assumed to be an infinite straight line of no thickness. If the wire possesses a small finite radius, an offset of the same radius applied to S can reduce the problem to the zero thickness line-accessibility problem we examine.

Now, reconsider S and let $S_i, i = 1, ..., m$ be the obstacles: these are additional (regular, smooth and closed) surfaces that take the roll of any occluding objects in the environment, such as other machine parts, etc. Hence, put in the above terminology, the visual hull of S is the smallest possible closed surface containing S with the property that *all of its points are LA within the given arrangement.*

Methods for computing the visual hull have been introduced, first for planar shapes [15, 10], for solids of revolution [9], for three dimensional solids with planar faces [10] and for solids with a generic smooth surface boundary [13, 3]. To the best of our knowledge, and while algorithm has been outlined for the smooth surface boundary case (for example in [13]), no implementation details are provided, such as actual computed visual hull examples and their computation times.

For now we assume that each of the surfaces (S and the obstacles) is entirely LA, had it been the *only* surface in the arrangement. In Section 6 we explain where this restriction originated and propose a possible solution. Along with a regularity assumption to be introduced in Section 3, the assumptions on the computational model enable a rigorous theoretic treatment and non-trivial examples, and is an important step toward a complete and practical solution to the line-accessibility and/or wire manufacturing of completely arbitrary and general input.

Our main contribution is a domain pruning algorithm that classifies most of the domain of S as either LA or LI with respect to the obstacles, without resorting to a full and expensive algebraic constraint solving, enabling the efficient computation of visual hulls from the fairly general class of parametric, smooth, piecewise polynomial (or rational) surfaces.

The rest of this paper is organized as follows: Section 2 reviews related work on the computation of visual hull, Section 3 provides the theoretic derivation of the algebraic constraints, as well as the visual hull construction algorithm. Section 4 describes the introduced domain pruning criteria, Section 5 presents several computed visual hull examples and finally Section 6 concludes and considers related future work.

2 Related Work

The visual hull has been mainly investigated with respect to problems involving three dimensional object reconstruction from two dimensional silhouettes and volume intersection. The precise definitions and well established theory are mostly due to A. Laurentini, who introduced the visual hull [9], and provided algorithms for its computation for objects of increasing difficulty, from planar shapes and

solids with planar faces [10], solids of revolution [12] and solids with a smooth surface boundary [13], where it is shown that the aspect graph [5] of the object can be used to compute the visual hull, by identifying specific visual events. Two sets of under-determined algebraic constraints are introduced in [13], which prescribe two types of univariate solutions. The first set describes all line-surfaces tri-tangency (simply tri-tangency hence after) events: triplets of points on the surfaces that share a common tangent line. The second set describes all line-surfaces bi-tangency (simply bi-tangency hence after) events: pairs of points on the surfaces that share a common tangent plane. As is shown in [13], these events contain the boundary between the LI/LA regions of the surface. We review the algebraic constraints in further detail in Section 3, where they are derived via an alternative approach, arriving at the same result. In [3], stricter necessary conditions for points to belong to the visual hull are presented, based on further curvature analysis at the neighborhood of the point. These conditions are in the form of further equality or inequality constraints, with case analysis involving knowledge of the Gaussian curvature.

Exploiting properties of B-spline surfaces, the criteria introduced in this work provide conditions for surface domain pruning based on the examination of the relative state of tangent planes and control points of the surface. Further, normal and tangent bounding cones plays a major role in this work, a frequently used tool such as in [16, 17, 8, 1] and more.

Piecewise smooth objects are treated in [4] by further deriving the aspect graph (where more visual events required). Computational issues such as the number of silhouettes required to reconstruct a three dimensional object are addressed in [11], and a computational geometric approach is taken in [15].

An essential capability for computing the visual hull of solids with smooth polynomial and/or rational parametric surface boundaries is algebraic constraints solving. We use the subdivision based solver [8] for fully determined systems, and [2] for under-constrained systems with a univariate solution.

3 Constraints Derivation and Post Filtering

This section portrays an alternative derivation to the two sets of algebraic constraints presented in [13]. As mentioned in Section 2, these two sets seek bi-tangency and tri-tangency events. Our interest shall be in a subset of their (univariate) solutions, as the curves inscribing the boundaries between the LA and the LI regions. Since the arguments in the current section do not distinguish between the objective surface S and the obstacles, for simpler notation we shall simply refer to the surface S.

The required terminology is now set, relying on standard concepts that can be found in differential geometry texts such as [6] or [14]. A precise definition of the decomposition of our interest is the following:

Definition 2. *We define an equivalence relation on S, by calling $P, Q \in S$ related if there exists a (continuous) path $\gamma : [a, b] \to S$ such that:*

1. *$\gamma(a) = P$ and $\gamma(b) = Q$.*
2. *Either $\gamma(t)$ is LA for all $t \in [a, b]$ or $\gamma(t)$ is LI for all $t \in [a, b]$.*

An equivalence class of the above relation is called a line-accessibility (LA) region *or a* line-inaccessibility (LI) region *(according to the common status of its points).*

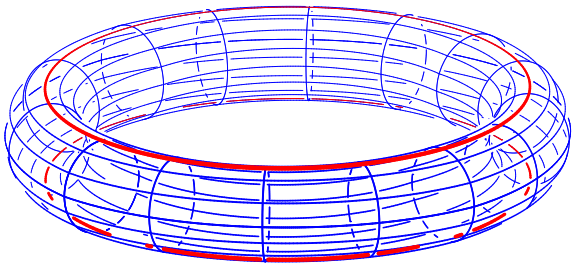


Figure 1: The parabolic lines (in red) of a torus present a simple case where a plane is in a tangential contact with the surface along a curve. Note the parabolic lines also serve as a (part of the) boundary between the LA and the LI regions of the torus.

In other words, an LI/LA region is a largest possible connected subset of S, such that all of its points have the same accessibility status. Note that an immediate consequence of Definition 1 is that the LA regions are closed subsets of S while the LI regions are open (relative to S, in the sub-space topology induced by $\mathbb{R}^3$). Next, consider the event of crossing from one such region to its neighbor: let $P, Q \in S$ be points such that there's a continuous path $\gamma : [a, b] \to S$ with $\gamma(a) = P$ and $\gamma(b) = Q$. Assume P is LA and Q is LI. Hence, for some value $\tilde{t} \in [a, b]$ the status must change for the first time: the minimal $t \in [a, b]$ for which $\gamma(t)$ is LA, is the event we are interested in, and is described in the following definition:

Definition 3. *A point $P \in S$ is said to belong to the boundary of the LA (and LI) regions, or simply* a boundary point, *if every neighborhood $U \subset S$ of P contains both an LA point and an LI point of S.*

The LA property of a point P can clearly be realized via many LA directions, and in fact via infinitely many (an entire sub-interval of $[0, 2\pi)$ of arguments in $T_P S$), or all directions, as is the case, for example, with any point of a sphere. On the other hand, an LI point has no such directions at all. Hence, it is natural to hope that the boundary points, as those taking the role of the transition location from "no directions" to "many directions", would be characterized by a unique LA direction (the first direction to be introduced at the entry point to the accessible region). This, however, is not true, as can be easily demonstrated by a standard torus, for example. At the entry point to the LA region, at the parabolic line of the torus, infinitely many LA directions are simultaneously introduced (See Figure 1).

We now defined the notion of a *blocked accessibility direction*:

Definition 4. *Let $v \in T_P S$ be an LA direction for $P \in S$ with an accessibility angle θ. The direction v is called* blocked *if, when rotated in $T_P S$ about P, for both clockwise and counter-clockwise arbitrarily small rotation - the rotated line is no longer an LA direction. In other words: for some $\epsilon > 0$, θ is the only accessibility angle in the interval $(\theta - \epsilon, \theta + \epsilon)$.*

Remark 5. *The direction with respect to which the argument angle θ is measured can clearly be chosen such that it varies continuously with respect to t, since S is C^1 smooth. For example, a possible choice is to measure θ against S_u.*

Armed with the notion of blocked accessibility, we now state a regularity assumption on (the geometry of) the surface, under which we may restrict our attention to points with blocked accessibility directions:

Definition 6. *The surface S is said to satisfy the* accessibility regularity assumption *(or simply the* regularity assumption *in this context) if for any plane that is tangent to S, tangency occurs at isolated points. Alternatively: for every $P \in S$, there's a positive radius r, such that for all $Q \in B(P,r) \cap S$, if $Q \neq P$ then $T_Q S \neq T_P S$.*

The typical situation avoided by the regularity assumption is a case where a plane is tangent to S at a curve (e.g. the parabolic lines of a torus), or, for that matter, at an entire planar region of S.

3.1 The Dynamic SSI Solution

The approach we now describe enables the theoretic derivation of the equations for tri-tangency and bi-tangency events by inspecting planar curve-line interactions. Let $\gamma : [a,b] \to S$ be a continuous path connecting $P = \gamma(a)$ and $Q = \gamma(b)$. For a fixed $t \in [a,b]$, consider the intersection:

$$(T_{\gamma(t)} S) \cap S. \tag{1}$$

This set is now viewed as the solution of an SSI problem of the two surfaces involved - the surface and its tangent plane. Allowing t to vary in $[a,b]$, we have a one parameter family of SSI problems, and consequently a one parameter family of solutions. Now, fix the point of view to that of a 2D observer located always at $\gamma(t)$. The plane is thus static - the frame of reference is chosen such that it is attached to the plane, and has its origin at $\gamma(t)$. The SSI solution, however, from this viewpoint, now has components that dynamically deform, appear, vanish, merge and split.

A solution point of an SSI problem can either be a transversal intersections or a tangency point of the two surfaces involved. By our assumption, points of the latter type are isolated. On the other hand - applying the regular level set theorem [14] we know that away from these isolated singularities, the solution is a one dimensional manifold. Finally, since one of the surfaces in the problem is compact - the solutions must be compact as well. Compact one-manifolds are closed curves. Isolated singularities can either belong or not belong to a solution curve. Hence, solution components of SSI problems that satisfy all our assumptions are limited to:

1. Simple closed loops.
2. Isolated solution points.
3. Curves that admit isolated self intersections.

By assumption, S is C^1 smooth and γ is continuous, hence $T_{\gamma(t)} S$ changes continuously[2] with t. Hence, the SSI solution is continuously deformed between those values of t that can be regarded as the topological events: The creation or disappearance of a component (which always occurs via a single point, by the regularity assumption) and the merge or split of existing components.

The preceding discussion and the regularity assumption on the surface enable us to interpret the LA property of $\gamma(t)$ with respect the line l_θ completely, by examining the interaction of l_θ with the SSI solution, in the tangent plane:

Proposition 7. *The point $\gamma(t)$ is LA via l_θ if and only if no points of l_θ are interior to the region bounded by any closed curve component of the SSI solution.*

[2] Formally, the t parameter family of tangent planes at the point $\gamma(t) \in S$ is given by parameterized tri-variate:

$$h(r,s,t) = S(\alpha(t)) + S_u(\alpha(t)) \cdot r + S_v(\alpha(t)) \cdot s,$$

where α is the respective path in the parameter domain, defined by $\gamma = S \circ \alpha$. The continuity with respect to t is guaranteed by the continuity of γ and of the derivatives S_u, S_v.

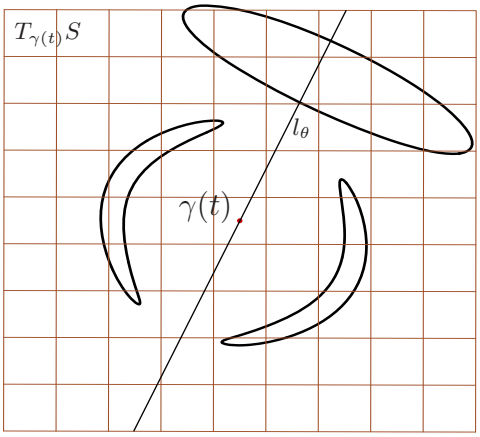


Figure 2: The LI status of $\gamma(t)$ via l_θ as reflected on $T_{\gamma(t)}S$: l_θ has interior points of a region inscribed by an SSI solution component.

Proof. The regularity assumption excludes tangency curves. Hence, crossing from the exterior to the interior (in the plane) of the region bounded by any closed curve component of the SSI solution is guaranteed to involve a switch in the status of interior/exterior to the volume bounded by S, in the $\mathbb{R}^3$ sense (Figure 2).

Since S is bounded, if l_θ has points that are interior to the region bounded by some closed curve component, l_θ must have points of both types: interior and exterior to the volume bounded by S. Hence it cannot be an accessible direction. Conversely, if it has no such interior points of a region bounded by a curve component, all of l_θ is of the same relative state to S in $\mathbb{R}^3$. Again, S is bounded, then this state must be all exterior, hence it is an accessible line. □

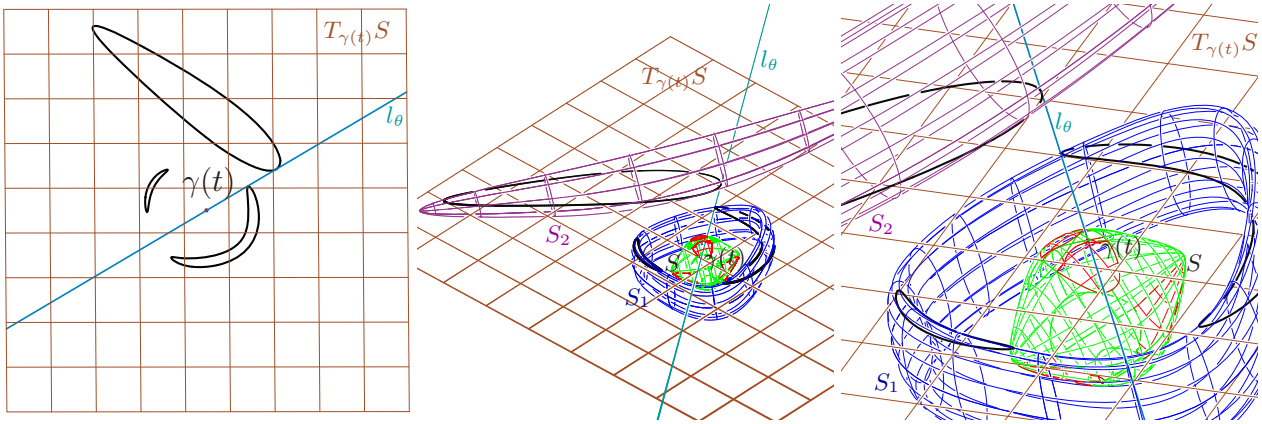
3.2 Boundary Points and Blocked Accessibility Directions

Fix a line l_θ and let t vary from $t = a$, at which l_θ has an interior point to a region bounded by some SSI solution curve, to $t = b$, at which it is strictly exterior to all components. The event at $\tilde{t} \in [a,b]$ which ceased the existence of an interior point implies that l_θ has at least one common point with the SSI solution (since it is not exterior) but no interior points. In other words: there's a rotation direction (either clockwise or counter-clockwise about $\gamma(t)$) that cannot be made without penetrating an SSI solution component by l_θ (Figure 3).

On the other hand - we claim that the other direction also causes penetration (Figure 4 (a)), given that $\gamma(\tilde{t})$ is a boundary point, namely $\tilde{t}$ is the minimal t value for which there exists a direction with no interior points, and this first accessible line direction is l_θ. To see this, if l_θ can be rotated for some finite angle - there's a direction $\theta + \epsilon$ or $\theta - \epsilon$ that is entirely in the exterior of any SSI component, hence it had no interior points for some $\overline{t} < \tilde{t}$, contradicting the minimality of $\tilde{t}$. The above discussion in fact proved the following property:

Proposition 8. *Under the regularity assumption and Definition 1, all boundary points are LA, and via blocked directions only.*

Remark 9. *We have implicitly made an important use of the regularity assumption here: the fact that closed curves cannot instantly vanish or admit a jump discontinuity in their*



(a) (b) (c)

Figure 4: A surface arrangement and corresponding SSI solution for a tri-tangency case, in the tangent plane (a) and in the corresponding Euclidean space (b). (c) is a zoom of (b). At the first LA point in (a), as we move along $\gamma(t)$, the direction l_θ (and in fact any direction that is accessible) must be blocked: both clockwise and counter-clockwise rotation about $\gamma(t)$ cause l_θ to penetrate the interior of a region bounded by SSI solution components (in the configuration depicted in this example - two disjoint components at distinct locations from $\gamma(t)$). In (b) and (c), the tangent line (in cyan) admits a tangency point with both obstacles (S_1 in blue and S_2 in magenta) as well as with the objective surface (S in green). The SSI solution curves are shown in black, and the red regions in S depicts the LI regions whereas in green, the LA regions of S are presented.

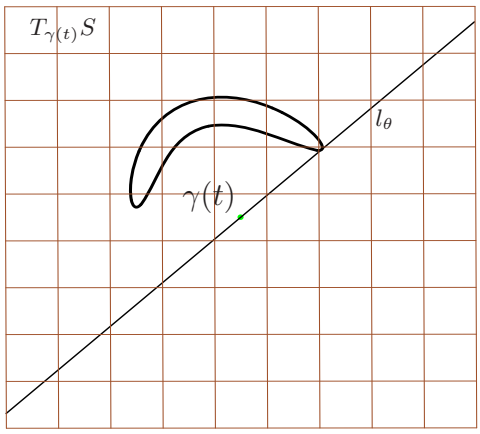


Figure 3: For a fixed direction l_θ, the transition of $\gamma(t)$ from LI via l_θ to LA via l_θ as reflected on $T_{\gamma(t)}S$: at least one rotation direction (counter-clockwise in this example) involves l_θ penetrating the interior of a region inscribed by an SSI solution component.

min/max argument angle, is enabled by the above property: a vanishing curve (or part of it) corresponds, in terms of the SSI problem, to a non-isolated set of tangency points, and hence violates the regularity assumption.

3.3 Algebraic Constraints for Blocked Directions

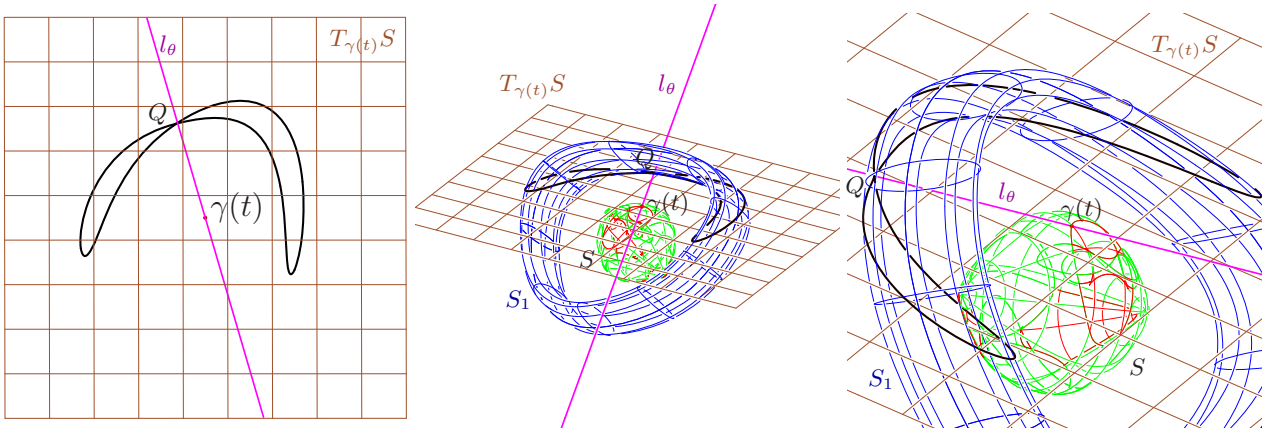
Finally, the notion of blocked accessibility directions enables us to prescribe the necessary algebraic conditions for a point to be a boundary point, from the planar SSI image we have described so far. We seek those events that cause a tangent line at $\gamma(t)$, in the plane, to be blocked. That is, it cannot be rotated clockwise or counter-wise without penetrating a closed curve component. The key idea is the following: to block the rotation of a line, *at most* two contact points with some SSI solution curve components are required. One is responsible for the blocking in the clockwise rotation direction and the other one for the counter-clockwise direction. It can happen that the two blocking contact points coincide, in which case the blocking effectively occurs via a single contact point distinct from $\gamma(t)$. It can also happen that the rotation is blocked when a single tangency occurs. However, these single tangency events do not contribute to the boundaries between the LA and LI regions as will be explained below. A line that is tangent to the surface can either have one, two or more than two tangency points. Thus, all possible line tangency cases of our interest are covered by the following:

1. *tri-tangency*: This is the generic case: $\gamma(t)$ as the center of rotation and two separate blocking contact locations (See Figure 4 (a)). Note that the rare and singular events of more than three tangency points are covered by finding all sub-triplets. Of course, there may be configurations with three or more tangency points such that only one rotation direction is blocked, and l_θ is not blocked. These will be pruned at post-filtering. Thus, we obtained the following set of constraints, for which we seek triplets of *pairwise distinct* points in the parameter domain $(u,v),(s,t),(a,b)$:

$$\begin{aligned}\langle S(u,v)-S(s,t),N(u,v)\rangle &= 0,\\ \langle S(u,v)-S(s,t),N(s,t)\rangle &= 0,\\ \langle S(u,v)-S(a,b),N(u,v)\rangle &= 0,\\ \langle S(u,v)-S(a,b),N(s,t)\rangle &= 0,\\ \langle S(a,b)-S(s,t),N(a,b)\rangle &= 0,\end{aligned} \tag{2}$$

where N is the (non unit) normal field of S, having five equations in six unknowns. These constraints enforces the orthogonality of the line l_θ through $S(u,v)$ and $S(s,t)$ and $S(a,b)$ to the normals of the surface, at those locations.

One should note that if all three locations are co-planar, Equations (2) are satisfied and yet the three surface locations need not be co-linear, a non-common case we purge at post-filtering. Alternatively, this could be avoided by requiring the three locations to be co-linear, by demanding that the appropriate cross product vanishes. We chose not to use this formulation since it implied higher degree constraints, and slowed down the computation in practice, despite the unwanted solution purging. An example for a tri-tangency event in the Euclidean space is depicted in Figure 4 (b) and (c).



(a) (b) (c)

Figure 5: A surface arrangement and corresponding SSI solution for a bi-tangency case, in the tangent plane (a) and in the corresponding Euclidean space (b), when l_θ is blocked by the same SSI solution component. (c) is a zoom of (b). The blocking of l_θ at a self intersection of point Q of an SSI solution component can occur, implying that the tangent planes at both locations ($\gamma(t)$ and Q) are the same. The tangent line in (b) and (c) (in black) admits a tangency point with the obstacle (S_1 in blue) and with the objective surface (S in green). The (self intersecting) SSI solution curve is shown in black. The red regions in S depicts the LI regions whereas in green, the LA regions of S are presented.

2. *bi-tangency*: This is the special case where the SSI component penetrated on clockwise and counter-clockwise rotation is the same component. Two components (and in fact the contact points) that were distinct in the tri-tangency case, are now the same. The event capable of introducing such configuration is when the line l_θ meets a self intersection point of an SSI solution, now denoted by Q (See Figure 5 (a)).

   Since self intersections are at singular SSI solution points (tangency points), we get an additional constraint: $T_{\gamma(t)}S = T_Q S$. Thus, we have the second set of constraints, seeking bi-tangency pairs that share a common normal direction:

$$\begin{aligned} \langle S(u,v) - S(a,b), N(u,v)\rangle &= 0, \\ \langle S_a(a,b), N(u,v)\rangle &= 0, \\ \langle S_b(a,b), N(u,v)\rangle &= 0, \end{aligned} \tag{3}$$

   where S_a and S_b are the partial derivatives of $S(a,b)$, having three equations in four unknowns. An example of a bi-tangency event in the Euclidean space is depicted in Figures 5 (b) and (c).

3. *uni-tangency*: We claim that we may ignore this event and do not attempt to formulate it algebraically. A trivial uni-tangency case is at points that are interior to the LA region, and recall we only seek the boundary between the LA and LI regions. However, a less trivial case is where there's a uni-tangency event that does admit both clockwise and counter-clockwise blocking of an accessibility line l_θ. Remember that although $\gamma(t)$ is, by assumption, an isolated tangency point, it may still be a member of a univariate SSI solution component. In such case l_θ is blocked, regardless of other events, by the SSI solution in the neighborhood of $\gamma(t)$. Consider, for example, a case where the only tangency location occurs at Q in Figure 5 (a). That said, if $\gamma(t)$ is the *only* tangency point along l_θ, l_θ does not contribute a line segment to the visual hull, namely: it does not create a ruling in any ruled surface patch that is intended to eventually replace an LI region. Hence, $\gamma(t)$ is not part of a curve bounding an LI/LA region, and uni-tangency cases of blocked directions are not of our interest.

3.4 Delineating the LA Regions

Recall that the tri-tangency and bi-tangency constraints give only a necessary condition for a point to be on the boundary of the LA regions, for several reasons:

1. It demands the existence of a blocked direction, whereas we seek points where *all* directions are blocked. Further, even if all directions are blocked, we have only proved that this is a necessary condition.

2. It looks for necessary blocking conditions, that may hold for l_θ and yet another occluding obstacle can still be punctured by l_θ.

3. In the tri-tangency case, there may be tri-tangent configurations that do not block l_θ. For example - if both tangency points other than $\gamma(t)$ have argument θ and both of the corresponding SSI solution components lay on the same half plane.

Hence, the solutions of the algebraic constraints in (2) and (3) give a preliminary decomposition of S, which is finer than required in most cases. Clearly all resulting sub-regions of S now consist of points sharing the same LA/LI status, but perhaps neighboring regions can be united - a process now briefly describe (remember that LA/LI regions are maximal connected subsets of the same status, as in Definition 2). A redundant curve arrangement in the parametric domain is depicted in Figure 6, for the surfaces in Figures 4 and 5. For each region we choose an interior point P and test its LI/LA state as follows: We solve the specific instance of the SSI problem determined by the respective tangent plane at P on S. Then, traverse all components of the univariate SSI solution, testing if there's an argument angle θ such that both θ and $\theta + \pi$ are not covered by the SSI components. If there's such an uncovered angle, the point P, and hence the entire region, is LA. Otherwise the region is LI. Finally, if two adjacent regions share the same LA/LI status, their common boundary can be pruned.

Solving the systems of equations we have derived is of great computational effort. Solving non-trivial examples (such as the ones used in this section) is greatly simplified by a surface domain pruning algorithm, which is the content of the coming Section 4.

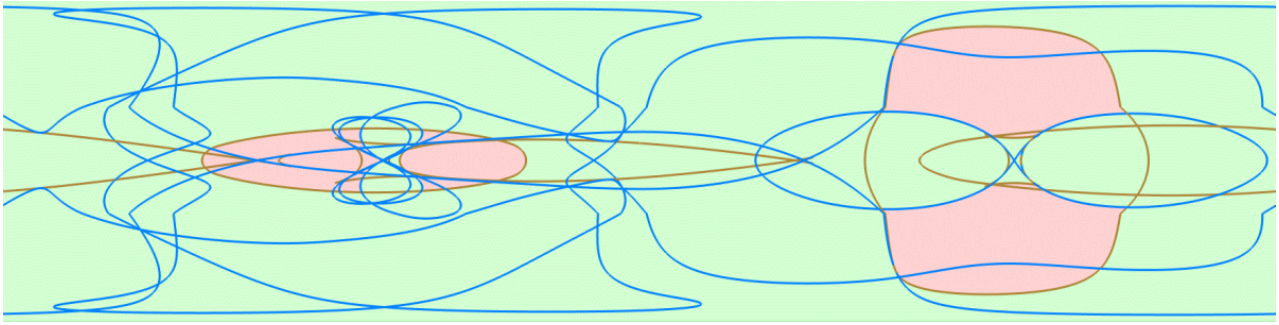


Figure 6: The LI (in red) and the LA (in green) regions as viewed in the parametric domain of the surface S. All the tri-tangency curves are shown in blue where as all bi-tangency curves are depicted in brown. The final LA regions are presented in green and the LI regions in red. For surface S also in Figures 4, 5 and 10.

4 Domain Purging

Denote the parametric domain of the surface S by:

$$D = [a, b] \times [c, d] \subset \mathbb{R}^2,$$

and let $D_i = [a_i, b_i] \times [c_i, d_i] \subset D$, be a sub-domain. This section provides two types of sufficient conditions regarding the LA/LI status of *all* points of $S(D_i)$. One detects situations where $S(D_i)$ is entirely LI, and the other detects situations where it is entirely LA.

The tests are applied recursively. First to the initial domain D, and then, as required, by means of a binary subdivision tree, until either all domains are successfully resolved (in which case - the entire problem is solved) or subdivision tolerance has been reached. In the latter case, only those sub-domains that remained (i.e. undecided by both tests) must be resolved via the next step - the algebraic constraints solution, as discussed in Section 3.

The size of the unclassified regions obtained by the domain purging step depends on the tolerances of two data structures. One is the domain subdivision tolerance. For the other we need to briefly recall the construction of the normal bounding cone and the tangent bounding cone of a parametric surface patch. All possible normal vectors of the surface patch $S(D_i)$ are given by the set:

$$N = \{S_u(u, v) \times S_v(u, v) : (u, v) \in D_i\}.$$

This yields a bivariate function $N(u, v)$ that can be bounded using its control points, assuming S is a B-spline or a NURBs surface. Denote by C^N, a cone that bounds the control points of N, with cone axis $\mathcal{A}$ and span angle α. A tangent bounding cone is denoted C^T and is given by a cone with same axis and the complementary span angle $\frac{\pi}{2} - \alpha$. C^T bounds all possible tangent directions to S. See Figure 7. Hence after, all cones we consider are double cones.

When the apex of C^T is translated to $P \in S(D_i)$, all tangent lines through P are bounded by C^T. That is, they are bounded not only as directions, but as lines in their Euclidean location. We divide C^T into n angular slices, denoted C^T_j, $j = 1, ..., n$, and properly extend each slice, now called $\overline{C^T_j}$, so that tangent lines of a specific argument angle domain shall be bounded by $\overline{C^T_j}$, in Euclidean space, for *all* $P \in S(D_i)$ (See Figure 8). To define the angular slices of C^T, and given n, the region C^T_j is defined by all tangent directions in C^T, such that when projected on the plane orthogonal to the axis of C^T, have arguments in the interval $I_j = [\frac{\pi j}{n}, \frac{\pi(j+1)}{n}]$ or in $I_j + \pi$.

The two subdivision tolerances (the domain subdivision tolerance and the value of n) can be understood as follows: the subdivision tolerance bounds regions of Euclidean positions, namely the locations tested for their LA/LI properties, whereas n bounds the directions from which accessibility is tested. When both tolerances are finitely small, our proposed pruning tests check whether a definite answer can be given to either one of two questions:

1. Are *all* points of $D_i \subset D$ mapped to points that are LA via *all* of their corresponding tangent directions in some $\overline{C^T_j}$?
2. Are *all* points of $D_i \subset D$ mapped to points that are LI via *all* directions in $\overline{C^T_j}$, for all j?

If the answer to either question is positive, $S(D_i)$ can be classified as fully LA or fully LI, respectively, and no further processing is necessary. The next sub-section describes how $\overline{C^T_j}$ is constructed.

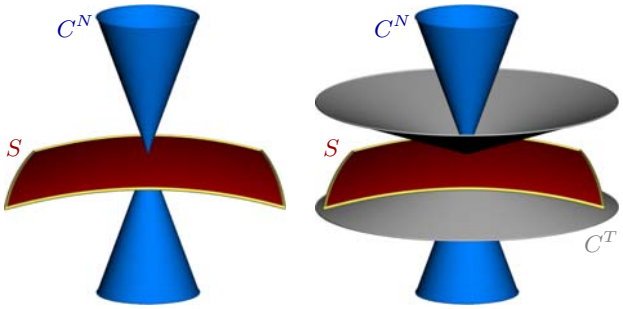
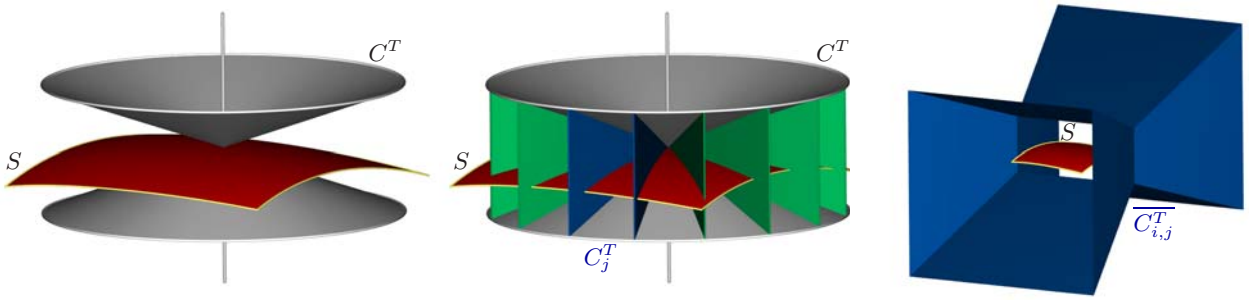


Figure 7: The normal (in blue) and tangent (in gray) cones of a surface patch (red). The normal cone alone is shown in (a), and both cones are shown in (b): the tangent cone is created by taking all possible orthogonal directions to the normal cone. Since the normal cone bounds all possible normal directions, the tangent cone bounds all possible tangent directions. When the apex of the tangent cone is located anywhere on the surface, the patch is fully contained in the tangent cone.

4.1 Construction of the Extended Angular Slice

Fix j and consider $C^T_j \subset C^T$. All tangent lines with arguments in I_j through a specific point P are bounded by C^T_j when its apex is translated to P. Denote the translated angular slice to P by $C^T_{j,P}$, and let $U^T_{i,j}$ be the union of $C^T_{j,P}$ over all $P \in S(D_i)$. Hence, $U^T_{i,j}$ contains all tangent lines through all $S(D_i)$ Euclidean location, and with arguments in I_j.

C^T_j can clearly be bounded by a double prism: two vertical planes bound the argument angles, while the other two, upper and lower, planes are determined by the apex and original cone, C^T, angle. Thus, C^T_j can be bounded by



(a) (b) (c)

Figure 8: The tangent cone C^T of $S(D_i)$ in (a) is divided to angular regions $\{C_j^T\}, j = 1, ..., n$ in (b). Each region C_j^T in (b) contains the directions with arguments in $\left[\frac{\pi j}{n}, \frac{\pi(j+1)}{n}\right]$ and their antipodal directions, in the plane orthogonal to the axis of C^T. In (c), region C_j^T is extended into double prism $\overline{C_{i,j}^T}$ by the bounding box of $S(D_i)$, capturing all tangents in C_j^T when they are repositioned as tangent lines through any point on $S(D_i)$, in Euclidean space.

a double prism now denoted R_j^T. Again, denote by $R_{j,P}^T$ the translated prism with an apex at P. Consequently, the union over all $P \in S(D_i)$ of $R_{j,P}^T$, denoted by $R_{i,j}^T$, contains $U_{i,j}^T$. Still, $R_{i,j}^T$ is difficult to compute, and we need a simpler bounding region.

Reconsider $S(D_i)$ and let $B_{Box}(S(D_i))$ the bounding box of surface region $S(D_i)$. The union over all $P \in B_{Box}(S(D_i))$ of $R_{j,P}^T$ clearly contains the union over all $P \in S(D_i)$ of $R_{j,P}^T$. Hence, we compute

$$\overline{C_{i,j}^T} = \cup_{P \in B_{Box}(S(D_i))} R_{j,P}^T,$$

by extending the double prism R_j^T around the center of $S(D_i)$ in x, y, and z, following by the sizes in $B_{Box}(S(D_i))$. In practice and assuming S is a B-spline or a NURBs surface, $B_{Box}(S(D_i))$ can be derived with ease, from the control points of $S(D_i)$. Again, see Figure 8.

4.2 The Sub-domain LA Test

We can now state the domain purging criteria:

Lemma 10. *Let D_i be a sub-domain for which we are testing the LA/LI status against potentially occluding region $S(D_k)$ and recall $\overline{C_{i,j}^T}$ is the extended double prism corresponding to D_i, with angular arguments in I_j. Then:*

1. *If $\overline{C_{i,j}^T} \cap \{int(S(D_k))\} = \phi$ then all points of $S(D_i)$ are LA with respect to $S(D_k)$.*
2. *Otherwise, $S(D_k)$ has a point inside $\overline{C_{i,j}^T}$. Let $C_{S(D_k)}^N$ be the normal cone of $S(D_k)$ and If $C_{S(D_k)}^N$ contains no direction that is orthogonal to a line in $\overline{C_{i,j}^T}$, then all lines of $\overline{C_{i,j}^T}$ are inaccessibility lines due to $S(D_k)$, for any point of $S(D_i)$.*

Proof. To see the first part, had there been an LI point $P \in S(D_i)$, all tangent lines through P would, by definition, have common points with the interior volume bounded by S. Specifically, those belonging to the angular slice $\overline{C_{i,j}^T}$ cannot be disjoint from $int\{S\}$, a contradiction.

As for the second part, for all points $P \in S(D_i)$ and all points $Q \in S(D_k) \cap \overline{C_{i,j}^T}$, line $\overline{PQ}$ is not orthogonal to $C_{S(D_k)}^N$. In other words, no line $\overline{PQ}$ can be tangent to C^1 surface $S(D_k)$ or $S(D_k)$ occludes the entire prism $\overline{C_{i,j}^T}$ and $S(D_i)$ is LI via the lines in extended prism $\overline{C_{i,j}^T}$ due to $S(D_k)$. □

Remark 11. *The above described process is not valid (i.e. might never succeed in deciding the LA/LI common status of the patch, no matter how small) if the surface patch self intersects. Again, we assume the surface is a regular two-manifold, and exclude such cases. Another assumption used here is that the curvature is bounded, and hence the patch is finitely small and has a valid normal cone. This is possible since we assume C^1 B-spline functions input. Even if not C^2 at knots, the one sided limits of the second derivatives exist and hence the curvature can only have jump discontinuities (therefore bounded). Between knots the functions are polynomials, implying the curvature is bounded, as it is continuous and defined on a compact domain.*

For the actual testing we use the control mesh for the surfaces involved, which is appropriate to guarantee the sufficient conditions. The computation amounts to several instances of testing point-plane relative states (for the first test) and simple overlapping tests for angular, planar regions for the second.

In summary, for a given domain D_i, traverse all extended angular cone slices until either a cone slice is found to be disjoint from the interior volume bounded by S, in which case $S(D_i)$ is entirely LA, or until the entire cone has been visited, in which case we engage the opposite test: traverse all angular cone slices until a slice cannot be guaranteed to be entirely composed of LI directions. If all slices have been visited, the patch $S(D_i)$ is completely LI. Otherwise, no definite conclusion could be made, and D_i is divided. This process results in a collection of sub-domains, in $\mathbb{R}^6$ for Equation (2) and $\mathbb{R}^4$ for Equation (3), to precisely trace the tri-tangency solutions and bi-tangency solutions, respectively. See Figure 9, for example.

5 Test Results

This section provides more examples of line accessibility computations. The surfaces in examples 1 to 4 are C^1, parametric B-spline surfaces, of order 4×3, with 50 to 70 control points each. Our first example is depicted in Figures 9 and 10. A general view of this first example can be seen in Figure 10 (a). To demonstrate the dramatic reduction of the domain due to the purging algorithm, Figure 10 (b) presents the result of the pruning step for this example. The vast majority of the surface region is classified (red and green) and only the zones near the boundary are left for further processing. Figure 9 shows a zoom-in on Figure 10 (b), and also depicts the division tree of the

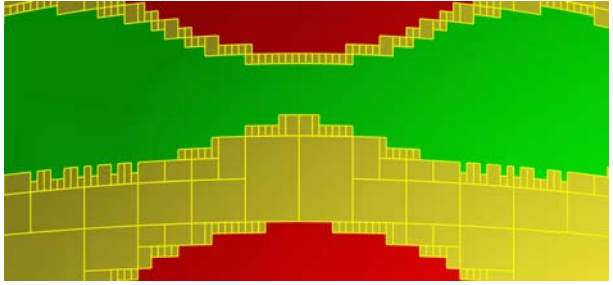


Figure 9: The domain pruning subdivision tree, zooming in on the surface region of Figure 10 (b).

domain pruning process. All regions that could be classified as LA are drawn in green and all regions that are classified as LI are presented in red. Only the remaining undecided regions, in yellow, are sent as input to the constraints solver [8, 2], yielding the exact boundary of the required regions, as is shown in Figure 10 (c).

Three more examples are presented in Figures 11, 12, and 13. A general view of the arrangement is depicted in (a) for all these examples, the domain pruning result is shown in (b) and the final precise LA/LI decomposition of S is depicted in (c).

Finally, Table 1 presents some timings and memory use results, measured on an Intel Xeon 3.2GHz PC with an 8GB of main memory, using a single core. These results clearly show the significant improvement due to the domain pruning process. Computation time is now reduced between one and two orders of magnitudes. As can also be seen from Table 1, the majority of the computation time is spend in the algebraic solver. Further inspection, not shown in the table, reveals that the solution of the tri-tangencies (Equation (2)) takes a significant portion of that time. This is not surprising due to the higher dimensionality of the problem compared to the bi-tangency case of Equation (3). Another noteworthy beneficial side effect of the domain pruning is the reduced memory overhead. From order of hundreds of megabytes before the pruning to a couple of dozens only.

It is interesting to observe the large difference in the timings, as used by the algebraic solver, in the example of Figure 12. This variance is due to singular (vanishing Jacobian) surface locations near tri-tangency solution points, that force many unnecessary subdivisions in the solver. These singularities obviously affect the relative benefits once can expect from the domain pruning. A special treatment might be due to such singular locations, in the future.

6 Conclusion and Future Work

In this paper, we have presented sufficient conditions for the line accessibility classifications of freeform surfaces, critical for wire EDM and hot-wire cutting, among others. Once the proposed domain pruning approach is applied, a significant reduction in the surface domains that required the use of the algebraic constraints is gained, enabling an efficient and tractable computation of the line accessibility query.

The boundary curves are traced via already known algebraic constraints, which have been reviewed here via an alternative theoretic derivation, using the dynamic SSI solution model. The solution presented does not handle self occlusions, and assumes that each surface in the arrangement is LA on its own. The reason can be found in Equations (2) and (3). If $S(u, v)$ is $S(a, b)$, any equations having the term $(S(u, v) - S(s, t)$ will vanish for $u = s$ and $v = t$.

Handling this self occlusion in Equations (2) and (3) can be approached via techniques such as self intersection elimination, such as in [7]. Alternatively, one should note that completely elliptic and completely hyperbolic patches with a bounded normal deviation must be completely LA with respect to themselves. Hence, geometric based division that ensures no self occlusions, possibly along parabolic lines can be exploited as well. Interestingly enough, the domain purging step presented in this work can possibly be augmented with a test for rejecting self occluding patches.

The regularity assumption made in Definition 6 is clearly strict and excludes valid geometry such as the torus in Figure 1. Ways to relax this regularity restrictions must be further investigated.

Another restriction is the C^1 smoothness of the input. A possible approach to handle surfaces with C^1 discontinuities, is to find the bi-tangency pairs and tri-tangency triplets that involve both curves and surfaces, when the curves are discontinuity curves that are always iso-parametric curves for B-spline surfaces. Further, computing bi- and tri-tangency lines when (discontinuity) curves are involved is a simpler problem compared to the bi- and tri-tangency lines of surfaces that are computed in this work, due to the lower dimensionality that is involved.

Another possible future step is the automatic completion of the LI (red) regions in the objective surface with ruled patched, constructing its complete visual hull with respect to the arrangement. The generation of wire-EDM (or hot wire cutting) tool-path for the LI regions is now reduced to devising a continuous tangent line sweep along those ruled surfaces. For the detected LA regions of the objective surface, the tool-path is not unique and is also an interesting future question to examine.

References

[1] G. Barequet and G. Elber. Optimal bounding cones of vectors in three dimensions. *Information Processing Letters*, 93(2):83–89, 2005.

[2] M. Bartoň, G. Elber, and I. Hanniel. Topologically guaranteed univariate solutions of underconstrained polynomial systems via no-loop and single-component tests. *Computer-Aided Design*, 43(8):1035–1044, 2011.

[3] A. Bottino and A. Laurentini. The visual hull of smooth curved objects. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(12):1622–1632, 2004.

[4] A. Bottino and A. Laurentini. The visual hull of piecewise smooth objects. *Computer vision and image understanding*, 110(1):7–18, 2008.

[5] K. W. Bowyer and C. R. Dyer. Aspect graphs: An introduction and survey of recent results. *International Journal of Imaging Systems and Technology*, 2(4):315–328, 1990.

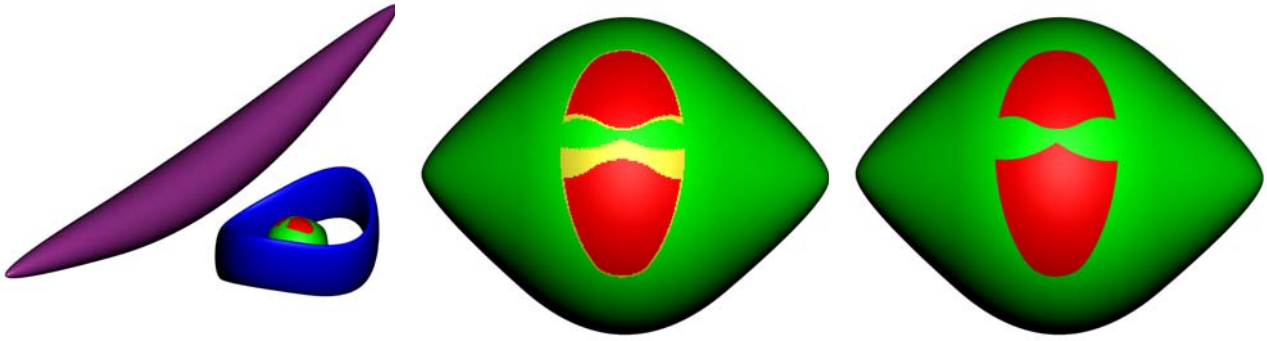
[6] M. P. Do Carmo. *Differential geometry of curves and surfaces*, volume 2. Prentice-Hall Englewood Cliffs, 1976.

[7] G. Elber, T. Grandine, and M.-S. Kim. Surface self-intersection computation via algebraic decomposition. *Comput. Aided Des.*, 41(12):1060–1066, Dec. 2009.

[8] G. Elber and M.-S. Kim. Geometric constraint solver using multivariate rational spline functions. In *Proceedings of the sixth ACM symposium on Solid modeling and applications*, pages 1–10. ACM, 2001.

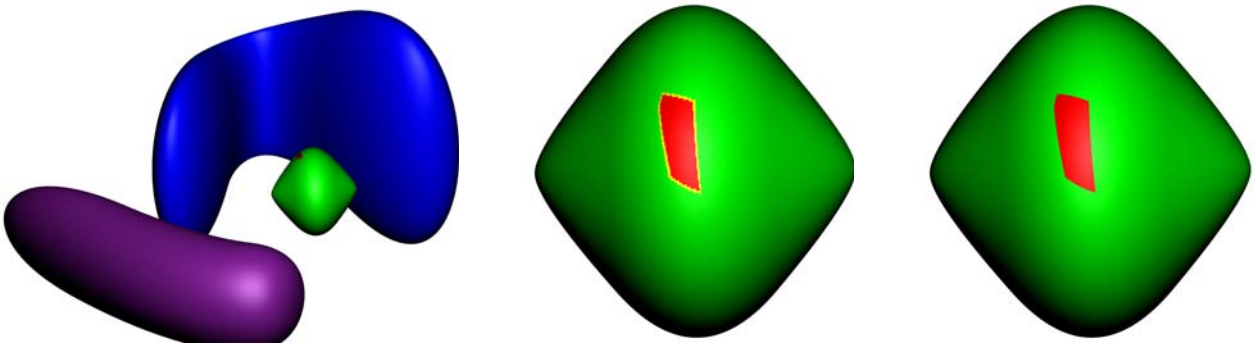
[9] A. Laurentini. The visual hull of solids of revolution. In *Pattern Recognition, 1992. Vol. I. Conference A: Computer Vision and Applications, Proceedings., 11th IAPR International Conference on*, pages 720–724. IEEE, 1992.

[10] A. Laurentini. The visual hull concept for silhouette-based image understanding. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 16(2):150–162, 1994.



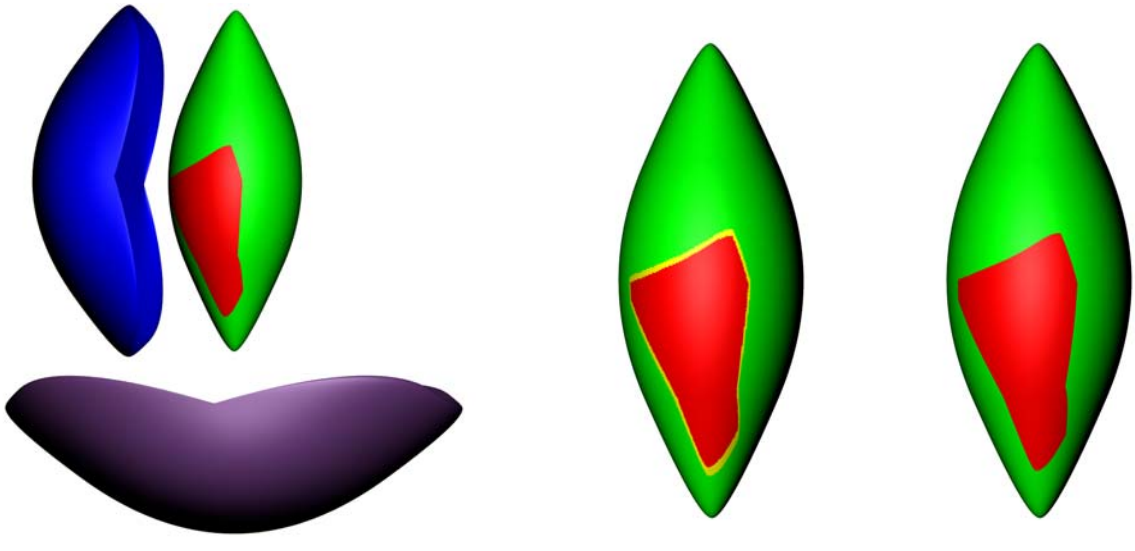
(a) (b) (c)

Figure 10: (a) An arrangement of the objective surface (green and red) and two obstacles (blue and magenta). The accessible regions are depicted in green and the inaccessible regions are shown in red. (b) Output of the domain purging algorithm, prior to applying the constraints solver. Fully classified areas are shown in green (LA) and in red (LI). Only the yellow regions are to be further classified, and are passed to the algebraic solver for precise analysis. (c) The precise boundary is traced using the algebraic constraints solver, fully determining the boundary between the LA and LI regions. See also Figure 9.



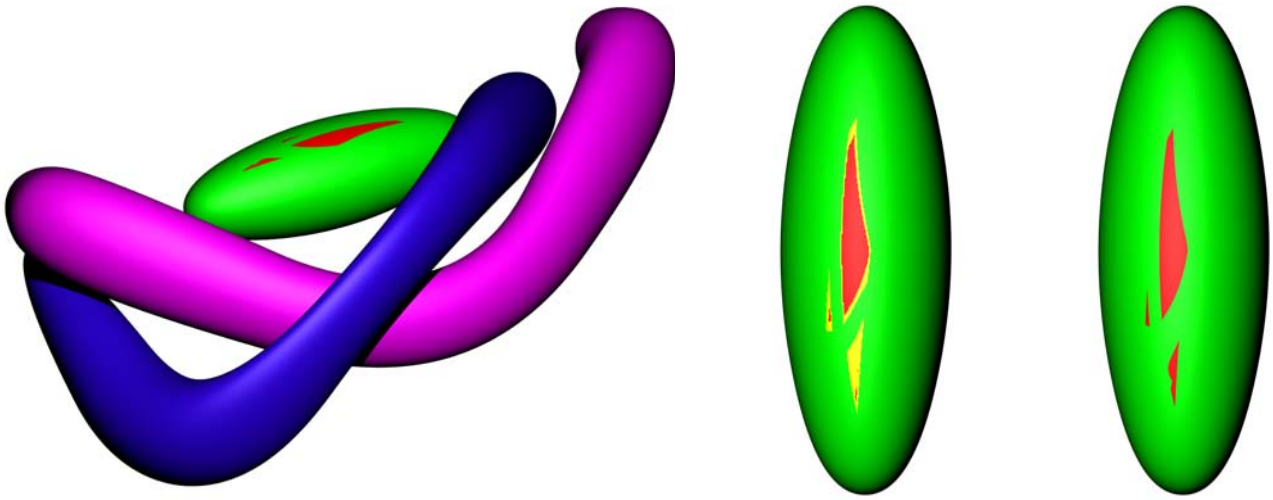
(a) (b) (c)

Figure 11: (a) An arrangement of the objective surface (LA regions in green and LI regions in red) and the obstacles (blue and magenta). (b) shows the different classified region after the domain pruning process while (c) presents the final result.



(a) (b) (c)

Figure 12: (a) An arrangement of the objective surface (LA regions in green and LI regions in red) and the obstacles (blue and magenta). (b) shows the different classified region after the domain pruning process while (c) presents the final result.



(a) (b) (c)

Figure 13: (a) An arrangement of the objective surface (LA regions in green and LI regions in red) and the obstacles (blue and magenta). (b) A zoom into the objective surface S.

	Time using Algebraic Solver Only		Time Using Pruning			
Example	Total Time (hh:mm:ss)	Memory Use (MBytes)	Pruning Time (hh:mm:ss)	Algebraic Solver Time (hh:mm:ss)	Total Time (hh:mm:ss)	Memory Use (MBytes)
Figure 10	10:02:29	275	00:04:28	00:38:58	00:43:26	28
Figure 11	13:42:10	245	00:00:21	00:19:47	00:20:08	28
Figure 12	18:37:37	510	00:02:36	05:56:35	05:59:11	25
Figure 13	11:51:43	436	00:01:09	00:20:35	00:21:44	28

Table 1: Timing and memory consumption of the different examples presented in this work, with and without domain pruning.

[11] A. Laurentini. How many 2d silhouettes does it take to reconstruct a 3d object? *Computer Vision and Image Understanding*, 67(1):81–87, 1997.

[12] A. Laurentini. Computing the visual hull of solids of revolution. *Pattern Recognition*, 32(3):377–388, 1999.

[13] A. Laurentini. The visual hull of curved objects. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 1, pages 356–361. IEEE, 1999.

[14] J. M. Lee. *Introduction to smooth manifolds*, volume 218. Springer, 2012.

[15] S. Petitjean. A computational geometric approach to visual hulls. *International Journal of Computational Geometry & Applications*, 8(04):407–436, 1998.

[16] T. W. Sederberg and R. J. Meyers. Loop detection in surface patch intersections. *Computer Aided Geometric Design*, 5(2):161–171, 1988.

[17] T. W. Sederberg and A. K. Zundel. Pyramids that bound surface patches. *Graphical Models and Image Processing*, 58(1):75–81, 1996.