Proceedings of TMCE 2014, May 19–23, 2014, Budapest, Hungary, edited by I. Horváth and Z. Rusák
© Organizing Committee of TMCE 2014, ISBN 978-94-6186-177-1

EXTENDING CONSTRUCTIVE SOLID GEOMETRY TO PROJECTIONS AND PARAMETRIC OBJECTS

George Tzoumas
CNRS UMR 5158, LE2I
University of Burgundy, France
George.Tzoumas@u-bourgogne.fr

Dominique Michelucci
CNRS UMR 5158, LE2I
University of Burgundy, France
Dominique.Michelucci@u-bourgogne.fr

Sebti Foufou
Qatar University, Qatar
sfoufou@qu.edu.qa

ABSTRACT

Constructive Solid Geometry (CSG) is a widely used representation scheme in Computer Aided Design and Manufacturing (CAD/CAM). Solids are represented as Boolean constructions via regularized set operators. CSG representations are essentially binary expression trees with non-terminal nodes representing operators and terminal nodes typically representing primitive solids (like spheres). Existing algorithms in the literature compute various properties of CSG sets like connectivity and topology. In this paper we extend traditional CSG trees to support the projection operator. Such an extension allows existing algorithms for CSG sets to work on a greater variety of sets, in particular parametric sets, which are extensively used in CAD/CAM systems. A geometric primitive may be defined in terms of a characteristic function, which can be seen as the zero-set of a corresponding system along with inequality constraints. To handle projections, we exploit the Disjunctive Normal Form, since projection distributes over union. To handle intersections, we introduce the notion of dominant sets that allows us to express intersections as disjoint unions. Finally, we introduce the join operator as a means to deal with primitives consisting of more than one manifold. Our algorithm, based on a traversal of the final expression tree, generates a set of geometric constraints in the form of equations and inequalities that express the same set, avoiding extra unknowns like Lagrange multipliers. We conclude with implementation notes and possible extensions.

KEYWORDS

geometric modeling, constructive solid geometry, projection, constraint solving, disjunctive normal form

1. INTRODUCTION

Constructive Solid Geometry (CSG) is one of the most widely used representation schemes in Computer Aided Design and Manufacturing (CAD/CAM) [7, 8]. Solids are represented as Boolean constructions via regularized set operators (union, intersection, difference). CSG representations are essentially binary expression trees where non-terminal nodes represent operators and terminal nodes typically represent primitive solids (like spheres, cones, cuboids).

Our goal is to extend the descriptive power of classical CSG by introducing the projection operator. This immediately allows us to deal with a greater variety of objects, *i.e.*, objects defined as projections of other objects or parametric objects, including those defined by extrusions and sweeps. We do so by describing the projected objects as classical CSG expressions in the projected (lower-dimension) space. We provide a method to automatically generate the corresponding expressions in this space.

The basic motivation behind our work is that existing algorithms in the literature compute various properties of CSG sets like connectivity [5] and topology [4]. Thus our method can extend existing algorithms allowing them to deal with a greater variety of geometric sets. We developed our approach in order to adapt the algorithms presented in [5, 4].

Almost no literature exists regarding projections in CSG. There is some discussion in [11], where it is mentioned that projections are non-trivial to handle and the author deals only with the projection of unions, since projections propagate over unions. In this paper we show how to deal with intersections, by transforming intersections into disjoint unions.

We represent a complicated geometric solid object as an expression tree consisting of classical CSG operators, as well as the projection operator. Recall that these operators are applied on geometric primitives, which are in fact leaves of the expression tree. Note that in CSG modeling we consider *regularized sets*. A regularized set is equal to the closure of its interior. As a consequence, a set and its complement share their boundary. A geometric primitive in $\mathbb{R}^d$ is represented as a manifold f in $(d+1)$-space $(x_1, \ldots, x_d, s)$ where s is the characteristic variable of the manifold. This way, the geometric primitive consists of interior points where the s-coordinate is negative and of exterior points where the s-coordinate is positive. The boundary case $s = 0$ corresponds (in general) to the boundary of the object. Thus the geometric primitive is essentially a solid in d dimensions. For example, manifold $f(x, y, s) = x^2 + y^2 - 1 - s = 0$ describes the unit disk as a paraboloid in 3-space. For every point $(\hat{x}, \hat{y})$ inside the unit disk, there exists $\hat{s} \leq 0$ such that $f(\hat{x}, \hat{y}, \hat{s}) = \hat{x}^2 + \hat{y}^2 - 1 - \hat{s} = 0$. Due to the regularized set condition, we consider inequalities $s \leq 0$ or $s \geq 0$ instead of their strict counterparts. This modeling via the characteristic variable is in accordance with the classical representation as a finite Boolean combination of semi-algebraic or semi-analytic sets of the form $\mathbf{x} \in \mathbb{R}^d,\ F(\mathbf{x}) \leq 0$, where $F : \mathbb{R}^d \to \mathbb{R}$. However in our approach, the use of characteristic variables proves more convenient. In sec. 3.1 we show how parametric solids can be represented as well. The characteristic variable implicitly defines a characteristic function:

Definition 1. *Given a d-dimensional workspace $\mathbb{R}^d$, a value is defined for the characteristic variable s of a set A for every point $\mathbf{x} \in \mathbb{R}^d$. If $\mathbf{x}$ lies in the interior of A then $s \leq 0$. If $\mathbf{x}$ lies outside A then $s \geq 0$.*

We consider each node of the expression tree separately in order to compute a *contributing primitive* for each point in the set. That is in the case of projections, every point is associated with a point in the higher dimension space. The tree is traversed and a set of simple subsystems with inequality constraints is generated. We denote this set as $F(\mathbf{x}; s)$, where s is the characteristic variable.

A geometric set is described in *Disjunctive Normal Form (DNF)* as a union of intersections of primitives. Having the union operator at the top level has several advantages as we show in the sequel. Such canonical forms are also considered in other algorithms dealing with CSG representations [9].

The paper is organized as follows. Section 2 presents the computation of the DNF for CSG operators. Section 3 deals with projections and parametric objects showing how to obtain a classical CSG expression in the lower-dimension space. Section 4 presents several applications of our approach. Our reference implementation is presented in section 5 and finally, in section 6 we conclude with discussion and future extensions.

2. CONSTRUCTIVE SOLID GEOMETRY OPERATIONS

2.1. Disjunctive Normal Form

A CSG formula can be converted to DNF by applying De Morgan's laws and by distributing $\cap$ over $\cup$ as follows. Let A, B be geometric primitives. Let P, Q_i be geometric sets defined by an expression tree.

1. A and $\neg A$ are in DNF.
2. $A \cup B$ and $A \cap B$ are in DNF.
3. $\neg(Q_1 \cup Q_2) = \neg Q_1 \cap \neg Q_2$.
4. $\neg(Q_1 \cap Q_2) = \neg Q_1 \cup \neg Q_2$.
5. $P \cap (Q_1 \cup Q_2 \cup \ldots \cup Q_n) = (P \cap Q_1) \cup (P \cap Q_2) \cup \ldots \cup (P \cap Q_n)$

We also merge consecutive binary operators of the same kind as follows:

6. $(Q_1 \cap Q_2) \cap Q_3 = Q_1 \cap Q_2 \cap Q_3$.
7. $(Q_1 \cup Q_2) \cup Q_3 = Q_1 \cup Q_2 \cup Q_3$.

By applying the above rules we end up having an expression tree where the topmost operator is $\cup$ and each operand subexpression contains only the $\cap$ operator applied to either a primitive or its complement. Note that conversion to DNF may result in an exponential explosion of the formula (*e.g.*, when computing the DNF of $(A_1 \cup B_1) \cap \cdots \cap (A_n \cup B_n)$). On the other hand, classical methods like propagation of bounding boxes [1, 2] can discard useless DNF clauses and reduce the number of nodes in the expression tree.

2.2. Complement

The simplest operation is the complement. If $A : f(\mathbf{x}; s) = 0$ then $\neg A : f(\mathbf{x}; -s) = 0$.

Remark 1. *It is important that the complement is still represented by points from the manifold.*

Recall that in Def. 1 we require the characteristic variable to be defined for all points in $\mathbb{R}^d$. For example, consider $f(x, y, s) = x^2 + y^2 - 1 + s^2 = 0$. Points outside the unit disk, like $(2, 2)$ are not represented at all, since $f(2, 2, s) = 0$ cannot be satisfied for $s \in \mathbb{R}$.

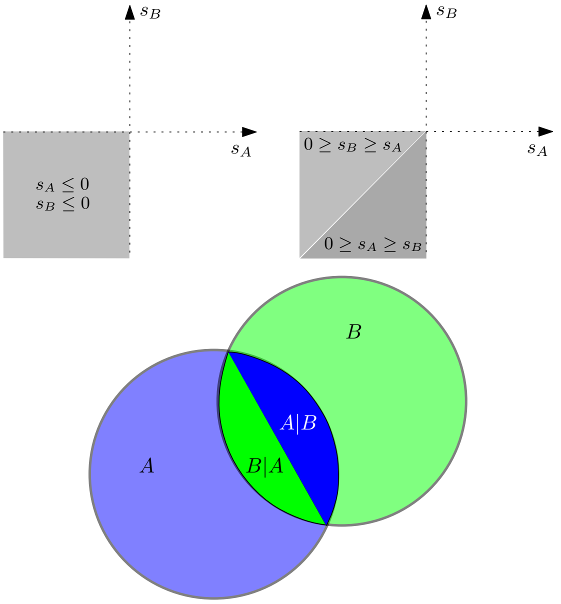

Figure 1 Top: Plot of the characteristic variables for $A \cap B = A|B \cup B|A$. Bottom: Visualization of $A, B, A|B, B|A$

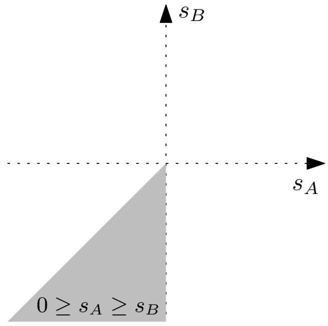

Figure 2 Plot of the characteristic variables for $\neg(A|B)$. The shaded area corresponds to $A|B$

2.3. Intersection

Let $P = A_1 \cap \ldots \cap A_n$ be an intersection of primitives. We have to impose the constraint that point $\mathbf{x}$ belongs to all A_i, $i = 1 \ldots n$ *at the same time*. This is achieved by setting the characteristic variable of the set to be $\max(s_1, \ldots, s_n)$. In order to represent $\max$, we consider all possible cases for $\max$.

Definition 2. *Let A, B_i be geometric sets and s_A and s_{B_i} their characteristic variables at point $\mathbf{x}$. Then:*

$$A|B_1, \ldots, B_n \;:=\; \mathbf{x} \in \mathbb{R}^d :\; 0 \geq s_A \geq s_{B_i},$$

$i = 1, \ldots n$, and we say that A dominates *$B_1, \ldots, B_n$.*

The notion of dominant set allows us to express intersections as unions. Note that operator $|$ has lower precedence than the comma.

Property 1. $A \cap B \;=\; A|B \;\cup\; B|A$

Proof. ($\Rightarrow$.) Let $\mathbf{x} \in A \cap B$. Then $0 \geq s_A$ and $0 \geq s_B$. If $0 \geq s_A \geq s_B$, then $\mathbf{x} \in A|B$. Otherwise, we have $0 \geq s_B \geq s_A$ therefore $\mathbf{x} \in B|A$. ($\Leftarrow$.) Let $\mathbf{x} \in A|B$. Then $0 \geq s_A \geq s_B$ which means $\mathbf{x} \in A \cap B$. Similarly when $\mathbf{x} \in B|A$. See Fig. 1. □

Property 2. $(A|B)|C \;=\; A|B, C$

Proof. $\mathbf{x} \in (A|B)|C \Leftrightarrow 0 \geq s_A \geq s_C \wedge 0 \geq s_A \geq s_B \Leftrightarrow \mathbf{x} \in A|B, C$. □

Property 3. $A|(B|C) \;\cup\; A|(C|B) \;=\; A|B, C$

Proof. $\mathbf{x} \in A|(B|C) \cup A|(C|B) \Leftrightarrow 0 \geq s_A \geq s_B \geq s_C \vee 0 \geq s_A \geq s_C \geq s_B \Leftrightarrow 0 \geq s_A \geq s_B \wedge 0 \geq s_A \geq s_C \Leftrightarrow \mathbf{x} \in A|B, C$. □

Property 4. $A \cap B \cap C \;=\; (A|B, C) \cup (B|C, A) \cup (C|A, B)$

Proof. $A \cap B \cap C = (A|B \cup B|A) \cap C = (A|B \cap C) \cup (B|A \cap C) = (A|B)|C \cup C|(A|B) \cup (B|A)|C \cup C|(B|A) = (A|B, C) \cup (B|C, A) \cup (C|A, B)$. □

Property 5. $\neg(A|B) \;=\; \neg A \;\cup\; \neg B \;\cup\; B|A$

Proof. Points in $A|B$ satisfy $0 \geq s_A \geq s_B$, therefore, the complement of this set consists of points where $s_A \geq 0$, $s_B \geq 0$ or $0 \geq s_B \geq s_A$. See Fig.2. □

2.4. Union

Let $P = A_1 \cup \ldots \cup A_n$ be a union of primitives. Since the formula is in DNF form, it suffices to consider each primitive separately, that is $A_i : f_i(\mathbf{x}; s_i)$, $i = 1 \ldots n$. Note that a point $\mathbf{x}$ may not be uniquely associated with a primitive, since it may belong to the intersection of many primitives, *i.e.*, the unions may not be disjoint. We can impose the uniqueness constraint by satisfying $s_i \leq s_j \forall i \neq j$, where A_j refers to each object in the considered intersection. Doing so can lead to similar properties like those in sec. 2.3.

3. PROJECTION

The first non-trivial operation which concerns sets that cannot be described with CSG primitives is *projection*. Let $A : F(\mathbf{x}; s)$ where $\mathbf{x} = (x_1, \ldots, x_d)$, then the projection of A with respect to x_i is denoted as $\pi_i(A)$. Let $\mathbf{x}^\mathbf{i} = (x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_d)$. Then $\pi_i(A) = F_\pi(\mathbf{x}^\mathbf{i}; s)$. Projections with respect to more than one dimension are denoted with commas, *e.g.* $\pi_{i,j}(A) = F_\pi(\mathbf{x}^{\mathbf{i,j}}; s) = \pi_j(\pi_i(A))$. When the particular dimension is not of importance we may simply write $\pi^2(A) = \pi(\pi(A))$. An interesting property of the projection is that it distributes over $\cup$. This is another argument in favor of the DNF form:

$$\pi(Q_1 \cup Q_2 \cup \ldots \cup Q_n) = \pi(Q_1) \cup \pi(Q_2) \cup \ldots \cup \pi(Q_n)$$

A naive way to deal with projections is to consider $F_{\pi_i} = F$, *i.e.*, just forget coordinate x_i. Doing so may allow for an 1-dimensional set of values for x_i and s, such that $F(\mathbf{x}; s)$ is satisfied. This may slow down an interval solver (used to find a cover of the set for example) since an infinite set of solutions will have to be covered. We fix this problem by introducing extra constraints so as to limit the range of the characteristic variable to a 0-dimensional set of values for every projected point. It is still possible that the extra constraints fail to reduce the dimension of the solution set, but this does not arise in practice.

A simple way to do that consistently is to choose the smallest value of the characteristic variable, among all possible values of coordinates $x_1, \ldots, x_k$ (the coordinates being eliminated). That is

$$\pi_{1,\ldots,k}(A) = \mathbf{x}^{\mathbf{1\ldots k}} \in \mathbb{R}^{d-k} : \begin{cases} \exists\, x_1, \ldots, x_k : \\ \mathbf{x} \in \mathbb{R}^d \cap A \\ s \text{ is minimal} \end{cases} \Leftrightarrow$$

$$F_\pi(\mathbf{x}^{\mathbf{1\ldots k}}; s) = \begin{cases} \exists\, x_1, \ldots, x_k : \\ F(\mathbf{x}; s) \\ s \text{ is minimal} \end{cases}$$

This way we pick the point that lies "deepest" in the set to map to the projected set. Note that the use of the term "minimal" in the above is abusive. We are actually looking for a *critical* point (without loss of generality). Thus, we don't have to perform extra computations to ensure that a critical point is actually a minimum. This is because we are interested in reducing the solution set to a hopefully 0-dimensional variety. It is perfectly acceptable for a point in the interior of the (projected) set to have not necessarily the smallest value of the characteristic variable, but some other (critical) value.

Note that coordinates $x_1, \ldots, x_k$ are no longer free variables, but take a value and become parameters. The minimization constraint can be written in terms of an optimization problem with constraints those exactly in F and the objective function s. Typical approach involves considering a Lagrangian (*i.e.*, the Fritz John conditions). This is quite powerful a technique, but it has the disadvantage that it introduces extra equations and unknowns. A more direct approach exists, which is equivalent to solving the Lagrangian system by hand and getting rid of redundant solutions. See Appendix A for an analysis of several Lagrangian systems. Here we make use of differential calculus and wedge products. For an introduction to wedge products and their applications in optimization problems the reader may refer to [15]. With $\longrightarrow$ we denote the sufficient constraints to describe a geometric set.

Theorem 1 (Projection of geometric primitive)**.** *Let $A : f(\mathbf{x}; s)$ be a geometric primitive. When projecting down k dimensions (eliminating $x_1 \ldots x_k$), the projection can be specified by:*

$$\pi^k(A) \longrightarrow \frac{\partial f}{\partial x_1} = \frac{\partial f}{\partial x_2} = \ldots = \frac{\partial f}{\partial x_k} = 0$$

Proof. $0 = ds \wedge df = ds \wedge (\frac{\partial f}{\partial x_1} dx_1 + \ldots + \frac{\partial f}{\partial x_k} dx_k + \frac{\partial f}{\partial x_s} dx_s) = \frac{\partial f}{\partial x_1} ds \wedge dx_1 + \ldots + \frac{\partial f}{\partial x_k} ds \wedge dx_k \iff \frac{\partial f}{\partial x_1} = \ldots = \frac{\partial f}{\partial x_k} = 0$. □

For an alternative proof using Lagrangians, see section A.

Note that we assume that there exists some critical value in the interior of the set, otherwise we would have to consider the boundary of the set. For complements, a critical value should exist in the exterior of the original set as well. We remedy this problem by considering the intersection with a big disk. This way we introduce a critical value where the characteristic variables of the set and the big disk are equal (to be explained in the sequel).

Definition 3. *Let A, B be geometric sets and s_A and s_B their characteristic variables at point $\mathbf{x}$. We define the* join *set $A \bowtie B$ as:*

$$A \bowtie B \;:=\; \mathbf{x} \in \mathbb{R}^d : \; s_A = s_B \;\wedge\; s_A \leq 0$$

We define the precedence of the new operators to be: $\neg \succ\, , \succ \mid \succ \bowtie \succ \cap \succ \cup$. Observe the similarity with the join operator from relational algebra. Indeed, we join the two relations $A(\mathbf{x}; s_A)$ and $B(\mathbf{x}; s_B)$ on their characteristic variable.

Definition 4. *We denote with $J_{i_1 i_2 \ldots i_n}(f_1, f_2, \ldots, f_n)$ the following $n \times n$ Jacobian determinant:*

$$\begin{vmatrix} \frac{\partial f_1}{\partial x_{i_1}} & \frac{\partial f_1}{\partial x_{i_2}} & \cdots & \frac{\partial f_1}{\partial x_{i_n}} \\ \frac{\partial f_2}{\partial x_{i_1}} & \frac{\partial f_2}{\partial x_{i_2}} & \cdots & \frac{\partial f_2}{\partial x_{i_n}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_{i_1}} & \frac{\partial f_n}{\partial x_{i_2}} & \cdots & \frac{\partial f_n}{\partial x_{i_n}} \end{vmatrix}$$

Lemma 1 (Projection of join sets)**.** *Let* $A : f_0(\mathbf{x}; s)$ *and* $B_i : f_i(\mathbf{x}; s), i = 1 \dots n$ *be geometric primitives. Then* $\pi^k(A \bowtie B_1 \bowtie \dots \bowtie B_n) \to$

$$\begin{cases} \emptyset, & k \le n \\ J_{i_0 i_1 \dots i_n}(f_0, f_1, \dots, f_n) = 0, & k > n \end{cases},$$

where $1 \le i_0 < i_1 < \dots < i_n \le k$.

Proof. Assume without loss of generality that we are projecting with respect to $x_1, x_2, \dots, x_k$. Considering the wedge product (to find the critical value of s) we have $ds \wedge df_0 \wedge df_1 \wedge df_2 \wedge \dots \wedge df_n = ds \wedge (\sum_{i=1}^k \frac{\partial f_0}{\partial x_i} dx_i + \frac{\partial f_0}{\partial s} ds) \wedge (\sum_{i=1}^k \frac{\partial f_1}{\partial x_i} dx_i + \frac{\partial f_1}{\partial s} ds) \wedge \dots \wedge (\sum_{i=1}^k \frac{\partial f_n}{\partial x_i} dx_i + \frac{\partial f_n}{\partial s} ds) = \epsilon_{x_{i_0} x_{i_1} \dots x_{i_n}} (\sum_{j=0}^n \frac{\partial f_j}{\partial x_{i_j}} dx_{i_j}) ds \wedge dx_{i_0} \wedge \dots \wedge dx_{i_n}$. Symbol ϵ is the permutation sign determined by the number of inversions in the considered permutation, which appears in the combinatorial definition of the determinant [14]. Now, if $k \le n$ the wedge product is identically zero, because of some dx_i being equal, due to the pigeonhole principle (we have a wedge product of $n+1$ factors with k choices for each factor, and we have that $dx_i \wedge dx_i = 0$). Otherwise, if $k > n$, the wedge product expands to $\binom{n+1}{k}$ coefficients which should all vanish. These coefficients are precisely $J_{i_0 i_1 \dots i_n}(f_0, f_1, \dots, f_n), 1 \le i_0 < i_1 < \dots < i_n \le k$. □

Since no extra condition is required to describe a projection of a join with respect to a single variable, we have that

Corollary 1. $\pi(A \bowtie B) = A \bowtie B$.

Theorem 2 (Projection of dominant set)**.**

$$\pi^k(A|B) = \pi^k(A)|B \cup \pi^k(A \bowtie B)$$

Proof.

First proof Without loss of generality we assume that we project with respect to $x_1, \dots, x_k$. Since we have a constrained optimization problem, the critical value can be attained either when a constraint is active or not.

$\pi^k(A|B) = \mathbf{x}^{\mathbf{1 \dots k}} \in \mathbb{R}^{d-k} : \exists\, x_1, \dots, x_k : (\mathbf{x} \in \mathbb{R}^d \cap A) \wedge (s_A \text{ is critical}) \wedge (s_A \ge s_B)$. This means that s_A takes its critical value on the critical points of $\pi(A)$ that happen to satisfy $s_A \ge s_B$, which is precisely $\pi^k(A)|B$ or somewhere where $s_A = s_B$, which is $\pi^k(A \bowtie B)$.

Second proof (Wedge product.) Let $A : f_0(\mathbf{x}; s)$ and $B : f_1(\mathbf{x}; s_1)$. We have a constrained optimization problem: Find $x_1 \dots x_k, s, s_1, u_1$ that optimize s (so that it achieves a critical value) subject to $f_0(\mathbf{x}; s) = f_1(\mathbf{x}; s_1) = g_1(s, s_1, u_1) = 0$, where $g_1(s, s_1, u_1) = s - s_1 - u_1^2$. Function g_1 enforces constraint $s \ge s_1$. Considering the wedge product to optimize s we have $ds \wedge df_0 \wedge df_1 \wedge dg_1 = 0 \Leftrightarrow ds \wedge (\sum_{i=1}^k \frac{\partial f_0}{\partial x_i} dx_i + \frac{\partial f_0}{\partial s} ds) \wedge (\sum_{i=1}^k \frac{\partial f_1}{\partial x_i} dx_i + \frac{\partial f_1}{\partial s_1} ds_1) \wedge (ds - ds_1 - 2u_1 du_1) = 0$. Now, if $k < 2$, we have that $0 = (\sum_{i=1}^k -2u_1 \frac{\partial f_0}{\partial x_i} \frac{\partial f_1}{\partial s_1} ds \wedge dx_i \wedge ds_1 \wedge du_1)$ which implies that $u_1 = 0$ or $\frac{\partial f_1}{\partial s_1} = 0$ or $\frac{\partial f_0}{\partial x_i} = 0, i = 1 \dots k$. If $k \ge 2$, then we have that $0 = (\sum_{i=1}^k -2u_1 \frac{\partial f_0}{\partial x_i} \frac{\partial f_1}{\partial s_1} ds \wedge dx_i \wedge ds_1 \wedge du_1) + (\sum_{i,j=1, i \ne j}^k -2u_1 \frac{\partial f_0}{\partial x_i} \frac{\partial f_1}{\partial x_j} ds \wedge dx_i \wedge dx_j \wedge du_1) + (\sum_{i,j=1, i \ne j}^k -\frac{\partial f_0}{\partial x_i} \frac{\partial f_1}{\partial x_j} ds \wedge dx_i \wedge dx_j \wedge ds_1) = 0 \Leftrightarrow u_1 \frac{\partial f_0}{\partial x_1} \frac{\partial f_1}{\partial s_1} = u_1 \frac{\partial f_0}{\partial x_2} \frac{\partial f_1}{\partial s_1} = \dots = u_1 \frac{\partial f_0}{\partial x_k} \frac{\partial f_1}{\partial s_1} = u_1 \sum_{i,j=1, i<j}^k [J_{ij}(f_0, f_1)]^2 = \sum_{i,j=1, i<j}^k [J_{ij}(f_0, f_1)]^2 = 0$. If $\frac{\partial f_1}{\partial s_1} = 0$, then we obtain $J_{ij}(f_0, f_1) = 0, 1 \le i < j \le k$. Otherwise, if $u_1 = 0$, then again $J_{ij}(f_0, f_1) = 0, 1 \le i < j \le k$. Otherwise we obtain that $\frac{\partial f_0}{\partial x_i} = 0, i = 1 \dots k$. In this case $J_{ij}(f_0, f_1)$ vanishes trivially.

The union of all previous cases is precisely $\pi^k(A)|B \cup \pi^k(A \bowtie B)$.

Third proof (Lagrangian.) See Appendix A.

□

Care has to be taken here that the set B in the expression $\pi^k(A)|B$ lies in a lower dimension. That is we consider points in $\pi^k(A)$ that happen to lie in B. We could denote B in this case as $B_{/\pi^k(A)}$ but we avoid so due to abuse of notation.

Let $[B_m]^{1:n}$ denote sequence $B_m, m = 1 \dots n$.

Theorem 3 (Projection of dominant sets, generalized)**.** $\pi^k(A|[B_m]^{1:n}) =$

$$\begin{array}{ll} & \pi^k(A)|[B_m]^{1:n} \\ \bigcup_{i=1}^n & \pi^k(A \bowtie B_i)|[B_m]^{1:n}_{m \ne i} \\ \bigcup_{i,j=1, i<j}^n & \pi^k(A \bowtie B_i \bowtie B_j)|[B_m]^{1:n}_{m \ne i, m \ne j} \\ \bigcup & \dots \\ \bigcup & \pi^k(A \bowtie B_1 \bowtie \dots \bowtie B_n) \end{array}$$

Proof. This comes as a generalization of Theorem 2. The same proof methods can be applied in this case with n constraints. □

3.1. Projection and parametric sets

Definition 3 implies that the join operator can be used to express the intersection of manifolds. In this case the join is performed on the common variables. For example $X : x - cos(t) = 0$, $Y : y - sin(t) = 0$. Now $X \bowtie Y$ expresses points (x, y) that lie on the unit circle. This way, the join variable t is implicitly eliminated. This is because we project with respect to t, due to the join operator being applied. Lemma 1 shows that projection of joins may be trivial if the number of variables projected is less than the number of terms in the join expression. Here $X \bowtie Y$ has 2 terms therefore, the resulting subspace consists of x and y only. More generically, given a parametric solid G in $\mathbb{R}^d$ defined by $X_i = f_i(\mathbf{x}; t_1, \dots, t_d)$, $i = 1 \dots d$, we can represent this set as the join of the defining manifolds. That is $G = X_1 \bowtie X_2 \bowtie \dots \bowtie X_d$. Assume that t_d is the characteristic variable. Now projection in the first $d-1$ dimensions eliminates the corresponding variables, and we have from Lemma 1 that $\pi^{d-1}(X_1 \bowtie \dots \bowtie X_d) = X_1 \bowtie X_2 \bowtie \dots \bowtie X_d$, since $k = d - 1 < d$. See sec. 4.3 for examples.

4. EXAMPLES

4.1. Projection of the unit sphere

Consider the unit sphere: $A : f(x, y, z, s) = x^2 + y^2 + z^2 - 1 - s = 0$. Then from Thm. 1 we have $F_\pi(x, y, s) =$

$$\begin{cases} f = x^2 + y^2 + z^2 - 1 - s = 0 \\ \frac{\partial f}{\partial z} = 2z = 0 \end{cases}$$

$$\Rightarrow \begin{cases} x^2 + y^2 - 1 - s = 0 \\ z = 0 \end{cases}$$

which is effectively the unit disk. Note that with the Lagrangian we have a bigger system: $F_\pi(x, y, s) =$

$$\begin{cases} x^2 + y^2 + z^2 - 1 - s = 0 \\ u_0 - v_1 = 0 \\ 2v_1 z = 0 \\ u_0 + v_1^2 = 1 \end{cases}$$

$$\Rightarrow \begin{cases} x^2 + y^2 - 1 - s = 0 \\ z = 0 \\ u_0 = v_1 \\ v_1 = \frac{-1+\sqrt{5}}{2} \end{cases}$$

4.2. Intersection of projections of intersection of spheres

Let E_1, E_2, E_3 be three spheres in $\mathbb{R}^3$, we want to express the object G defined as

$$G = \pi(E_1 \cap E_2) \cap \pi(E_1 \cap E_3).$$

The above expression will be transformed in DNF. We have that: $G = [\pi(E_1|E_2) \cup \pi(E_2|E_1)] \cap [\pi(E_1|E_3) \cup \pi(E_3|E_1)] = [\pi(E_1|E_2) \cap \pi(E_1|E_3)] \cup [\pi(E_1|E_2) \cap \pi(E_3|E_1)] \cup [\pi(E_2|E_1) \cap \pi(E_1|E_3)] \cup [\pi(E_2|E_1) \cap \pi(E_3|E_1)] = [\pi(E_1)|E_2 \cup E_1 \bowtie E_2] \cap [\pi(E_1)|E_3 \cup E_1 \bowtie E_3] \cup \cdots = [\pi(E_1)|E_2 \cap \pi(E_1)|E_3] \cup \cdots = [\pi(E_1)|E_2, \pi(E_1)|E_3)] \cup [\pi(E_1)|E_3, \pi(E_1)|E_2] \cup \cdots = \bigcup_{i=1}^{18} S_i$. That is G is equal to the union of 18 sets $S_1, \dots, S_{18}$ which in fact can be grouped into five sets depending on the contributing set being $\pi(E_1), \pi(E_2), \pi(E_3), E_1 \bowtie E_2, E_1 \bowtie E_3$, as shown in the following table:

set	contributing set	formula
S_1	$\pi(E_1)$	$\pi(E_1)\|E_2, \pi(E_1)\|E_3$
S_2		$\pi(E_1)\|E_3, \pi(E_1)\|E_2$
S_3		$\pi(E_1)\|E_2, E_1 \bowtie E_3$
S_4		$\pi(E_1)\|E_3, E_1 \bowtie E_2$
S_5		$\pi(E_1)\|E_2, \pi(E_3)\|E_1$
S_6		$\pi(E_1)\|E_3, \pi(E_2)\|E_1$
S_7	$\pi(E_2)$	$\pi(E_2)\|E_1, \pi(E_1)\|E_3$
S_8		$\pi(E_2)\|E_1, E_1 \bowtie E_3$
S_9		$\pi(E_2)\|E_1, \pi(E_3)\|E_1$
S_{10}	$\pi(E_3)$	$\pi(E_3)\|E_1, \pi(E_1)\|E_2$
S_{11}		$\pi(E_3)\|E_1, E_1 \bowtie E_2$
S_{12}		$\pi(E_3)\|E_1, \pi(E_2)\|E_1$
S_{13}	$E_1 \bowtie E_2$	$E_1 \bowtie E_2\|(\pi(E_1)\|E_3)$
S_{14}		$E_1 \bowtie E_2\|E_1 \bowtie E_3$
S_{15}		$E_1 \bowtie E_2\|(\pi(E_3)\|E_1)$
S_{16}	$E_1 \bowtie E_3$	$E_1 \bowtie E_3\|(\pi(E_1)\|E_2)$
S_{17}		$E_1 \bowtie E_3\|E_1 \bowtie E_2$
S_{18}		$E_1 \bowtie E_3\|(\pi(E_2)\|E_1)$

Let (x, y, z, r) denote a sphere centered at (x, y, z) with radius r. If $E_1 = (0, 0, 0, 1)$, $E_2 = (\frac{1}{2}, 0, \frac{1}{2}, \sqrt{\frac{3}{2}})$ and $E_3 = (-\frac{3}{2}, 0, \frac{3}{2}, \frac{3}{2})$ then $\pi(E_1 \cap E_2)$, $\pi(E_1 \cap E_3)$ and $\pi(E_1 \cap E_2) \cap \pi(E_1 \cap E_3)$ are shown in Fig. 3 and Fig. 4, bottom. Also visible are $E_1 \bowtie E_2$, $\pi(E_1)|E_2$ and $E_1 \bowtie E_3$ (Fig. 4, top).

4.3. Parametric disk in $\mathbb{R}^2$

Let $X(t, r) = x - r \cos t$ and $Y(t, r) = y - r \sin t$. With $t \in [-\pi, \pi)$ and $r \in [0, 1]$ we obtain the unit disk. However, we would like one parameter to be a characteristic variable. That is, negative values correspond to points in the interior of the solid, and positive values correspond to points in the exterior of the solid (and should cover the complement of the solid). We set $r = \sqrt{1 + s}$. Now our solid becomes

$$\begin{bmatrix} X(x, y, t, s) \\ Y(x, y, t, s) \end{bmatrix} = \begin{bmatrix} x - \sqrt{1+s} \cos t \\ y - \sqrt{1+s} \sin t \end{bmatrix}$$

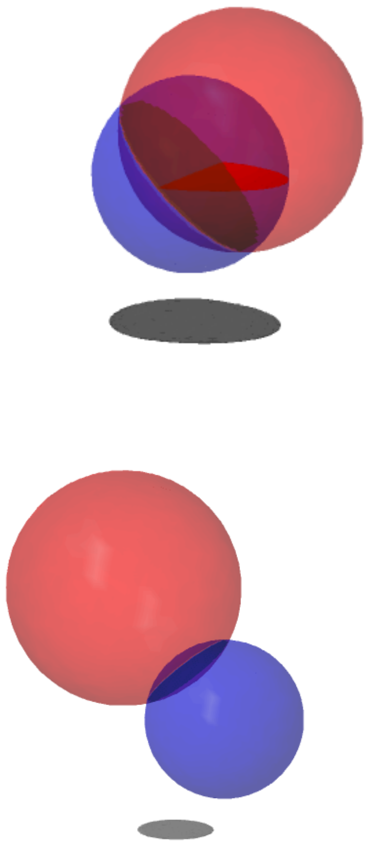

Figure 3 Top: $\pi(E_1 \cap E_2)$ in 3D. Bottom: $\pi(E_1 \cap E_3)$ in 3D

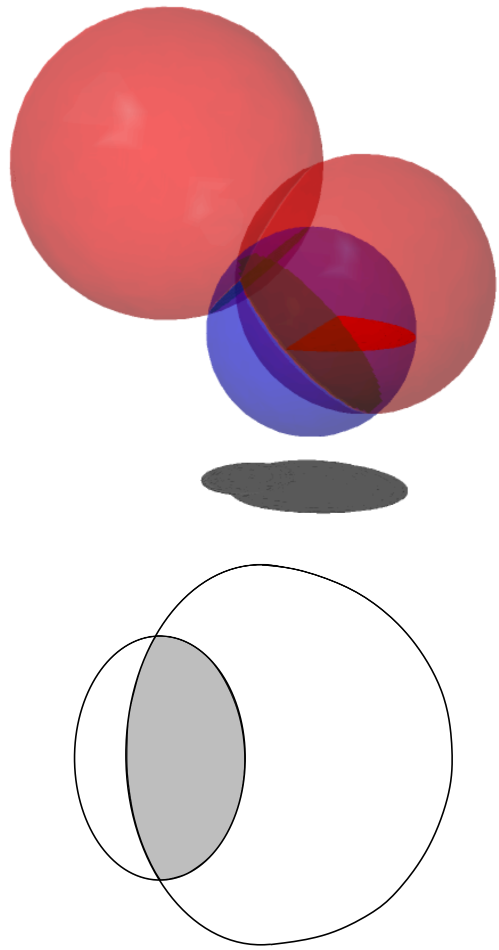

Figure 4 Top: $\pi(E_1 \cap E_2) \cup \pi(E_1 \cap E_3)$ in 3D; Bottom: $\pi(E_1 \cap E_2) \cap \pi(E_1 \cap E_3)$ in 2D

which is represented by $X \bowtie Y$ and yields the following constraints:

$$\begin{cases} x - \sqrt{1+s}\ \cos t & = 0 \\ y - \sqrt{1+s}\ \sin t & = 0 \end{cases}$$

The use of square roots may lead to problems, depending on the nature of the solver used. Moreover it leads to problems when one tries to compute the projection of the above set, because the square root function is not defined in $\mathbb{R}$, but only in $[0, \infty)$ (*e.g.*, where does the critical point of s lie?). It is possible to avoid using square root expressions at the cost of introducing one extra equation and one unknown. We consider $R(r, s) = r(r-1) - s$. Now our solid becomes $R \bowtie X \bowtie Y$. The join expression contains three terms, therefore it is equivalent to the projection with respect to two variables r, t yielding a parametric solid in (x, y, s), according to Lemma 1.

4.4. Parametric annulus in $\mathbb{R}^2$

As in the previous section, we set $X(t, r) = x - (\frac{1}{2} + r)\ \cos t$, $Y(t, r) = y - (\frac{1}{2} + r)\ \sin t$, $R(r, s) = r(r-1) - s$. Then $\pi_{r,t}(R \bowtie X \bowtie Y)$ is a 2D annulus in the xy-space, as shown in Fig. 5 bottom. Fig. 5 top shows the parametric construction $\pi_r(R \bowtie X \bowtie Y)$ in 3D (xyt-space) before being projected down with respect to t-axis. It is an infinite spiral ribbon along the t-axis (shown clipped).

5. IMPLEMENTATION

We have implemented the approach presented in this paper in Python/SAGE [10]. It automatically generates the necessary conditions (as subsystems) to describe the geometric set, which are then passed to Quimper for solving. The routine to transform the expression tree in DNF turns out to be non-trivial to implement, if one allows for simplifications and cancellations. Therefore the DNF may not be computed automatically for arbitrary complex trees, but it is easy to manually specify the corresponding subexpressions

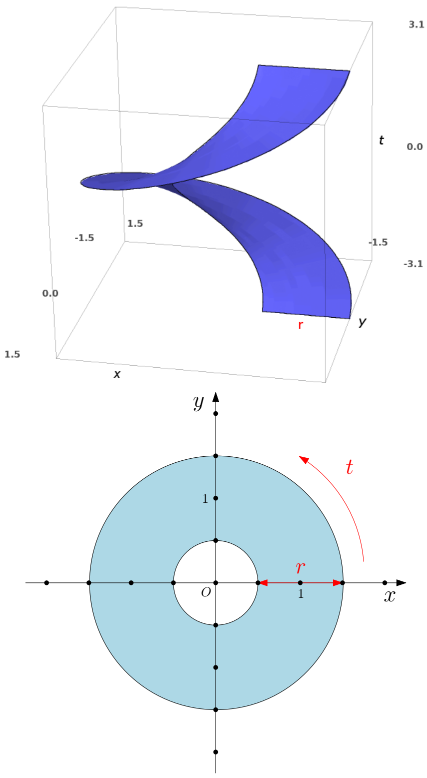

Figure 5 Top: Visualization of $\pi_r(R \bowtie X \bowtie Y)$ in 3D; Bottom: $\pi_{r,t}(R \bowtie X \bowtie Y)$ in 2D

in DNF. The expression tree is described by object constructors. For example, given spheres A and B, $\pi(A \cap B)$ is expressed as:

```
x,y,z = SR.var('x,y,z')
A=PrimitiveSet((x-3/4)^2+(y-3/4)^2+(z-3/4)^2<2/3,
    {x:RIF(-2,2), y:RIF(-2,2),z:RIF(-2,2)})
B=PrimitiveSet((x-1/4)^2+(y-1/4)^2+(z-1/4)^2<1,
        {x:RIF(-2,2), y:RIF(-2,2),z:RIF(-2,2)})
G=ProjectionSet(IntersectionSet(A,B),set([z]))
```

The output is the DNF expression: $\pi(A)|B \cup \pi(A \bowtie B) \cup \pi(B)|A \cup \pi(B \bowtie A)$. Note that although $\pi(B \bowtie A)$ is identical to $\pi(A \bowtie B)$ it still appears in the expression. We hope to allow for such optimizations in future versions. The four systems generated are:

1. $$\begin{cases} \frac{1}{16}(4z-3)^2 + \frac{1}{16}(4y-3)^2 + \\ \quad +\frac{1}{16}(4x-3)^2 - s_0 - \frac{2}{3} = 0 \\ \frac{1}{16}(4z-1)^2 + \frac{1}{16}(4y-1)^2 + \\ \quad +\frac{1}{16}(4x-1)^2 - s_1 - 1 = 0 \\ 2z - \frac{3}{2} = 0, \quad s_0 - s_1 \geq 0 \end{cases}$$

2. $$\begin{cases} \frac{1}{16}(4z-3)^2 + \frac{1}{16}(4y-3)^2 + \\ \quad +\frac{1}{16}(4x-3)^2 - s_0 - \frac{2}{3} = 0 \\ \frac{1}{16}(4z-1)^2 + \frac{1}{16}(4y-1)^2 + \\ \quad +\frac{1}{16}(4x-1)^2 - s_1 - 1 = 0 \\ s_0 - s_1 = 0 \end{cases}$$

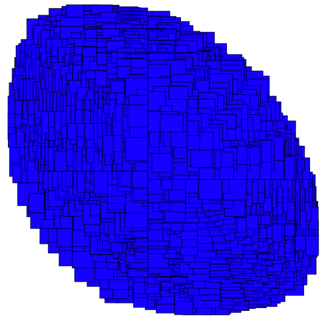

Figure 6 Plot of $\pi(A \cap B)$ in (x, y)

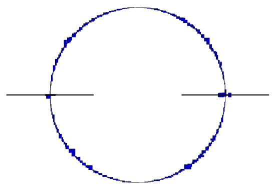

Figure 7 Plot of $\pi_{r,t,y}(R \bowtie X \bowtie Y)$

3. $$\begin{cases} \frac{1}{16}(4z-3)^2 + \frac{1}{16}(4y-3)^2 + \\ \quad +\frac{1}{16}(4x-3)^2 - s_0 - \frac{2}{3} = 0 \\ \frac{1}{16}(4z-1)^2 + \frac{1}{16}(4y-1)^2 + \\ \quad +\frac{1}{16}(4x-1)^2 - s_1 - 1 = 0 \\ 2z - \frac{1}{2} = 0, \quad s_1 - s_0 \geq 0 \end{cases}$$

4. Identical to 2.

Finally, the systems are solved with Quimper and the results are merged and plotted (Fig. 6).

As a second example, we present the code for the projection of the parametric annulus of Sec. 4.4 with respect to t, r, y.

```
x,y,t,r,s = SR.var('x,y,t,r,s')
R=PrimitiveSet(s-r*(r-1),
 {r:RIF(0,1),s:RIF(-2,2)},s)
X=PrimitiveSet(x - (1/2*cos(t) + r*cos(t)),
 {x:RIF(-2,2),t:RIF(-3.15,3.15),r:RIF(0,1)},None)
Y=PrimitiveSet(y - (1/2*sin(t) + r*sin(t)),
 {y:RIF(-2,2),t:RIF(-3.15,3.15),r:RIF(0,1)},None)
G=ProjectionSet(JoinSetMulti([R,X,Y]),[t,r,y])
```

The generated constraints are solved with Quimper and plotted in Fig. 7. The plot is shown as a 2D shape, however it represents an 1-dimensional object (the projection along the x-axis). It is evident that the projection is equal to the x range $[-\frac{3}{2}, \frac{3}{2}]$. The y-values shown are contributing points from the higher dimension, *i.e.*, the y-values that correspond to critical values of the characteristic variable s. According to Lem. 1, an additional Jacobian constraint has been taken into account, since the number variables projected is greater than or equal to the number of terms in the join expression:

$$\begin{vmatrix} 0 & 1-2r & 0 \\ (\frac{1}{2}+r)\sin t & -\cos t & 0 \\ -(\frac{1}{2}+r)\cos t & -\sin t & 1 \end{vmatrix} = 0$$

6. DISCUSSION

We have presented a method to extend classical CSG constructs with the projection operator. We exploited the DNF form, the join operator and the notion of dominant set. This is based on the observation that in general, only one geometric primitive should be contributing to each point the projection which lies in a lower-dimension space. The join operator deals effectively with boundary conditions where more primitives contribute to that point.

The interested reader may find an elaborate report on some preliminary work that led to this paper in [13, 12]. These works study the simple idea of directly describing the set with a set of equations and inequalities, possibly introducing (nested) optimization problems with the aid of Lagrange multipliers in the case of projections. However, merely considering a set of semi-algebraic equations (to be used with some black-box solver) is not efficient. Consider, for example, n unit disks centered at (x_i, y_i), $i = 1 \dots n$. Construct the $(n+1) \times (n+3)$ system consisting of the n equations $(x - x_i)^2 + (y - y_i^2) - 1 - s_i = 0$, $i = 1 \dots n$ as well as the equation $\prod_{i=1}^{n} (s - s_i) = 0$. Now the zero-set of the system with respect to x, y, s, s_i where $s \leq 0$ describes the union of the n disks. This naive approach yields an $(n+1) \times (n+3)$ system, while the same set can be expressed by concatenating the solutions of n independent equations in 3 variables. Extensive benchmarks have shown that such an approach is non-practical even when state-of-the-art solvers like Quimper [3] are considered. This led us to this paper's approach which manipulates the expression tree and avoids introducing Lagrange multipliers.

Other types of sets such as extrusions or sweeps should be fairly easy to be expressed in our framework, since extrusions and sweeps are parametric objects. For Minkowski sums things are more difficult, as we currently know of no easy way to describe the characteristic function (simply considering the sum of the characteristic variables is not enough). This will be a topic of future research. Another interesting problem is to study the complement of projections, which boils down to dealing with complements of join sets. The definition of $\neg(A \bowtie B)$ raises interesting questions. For example, let A, B be two spheres. Then $\neg(A \bowtie B) = \neg A \bowtie \neg B$, but there are cases (*e.g.*, when considering $\neg(A \bowtie \neg B)$) where the characteristic variable does not span both interior and exterior parts.

Extending the CSG representation is essential to extend geometric and topological algorithms. A basic motivation behind our approach is to extend the HIA method (Homotopy via Interval type Analysis) of [4] to objects more general than CSG like the aforementioned ones. The basic predicate of that algorithm is the star test, which determines if a set is homotopy equivalent to a point. Since our method expresses the projected object as classical CSG, it remains to implement the star test for the new objects (in the lower dimension space). This is a topic we are currently investigating.

ACKNOWLEDGMENT

This work has been funded by NPRP grant number NPRP 09-906-1-137 from the Qatar National Research Fund (a member of The Qatar Foundation). The statements made herein are solely the responsibility of the authors.

A. ALTERNATIVE PROOFS WITH LAGRANGE MULTIPLIERS

Assume that F contains k intermediate variables s_1, s_2, ..., s_k, n equations $f_j(\mathbf{x}; s_1 \dots s_k, s)$, $j = 1 \dots n$ and m inequalities $g_j(\mathbf{x}; s_1 \dots s_k, s) \leq 0$, $j = 1 \dots m$. Then, by using the Fritz John conditions (*e.g.*, [6]) we obtain $F_\pi(\mathbf{x^i}; s) =$

$$\left\{ \begin{array}{ccc} u_0 \nabla s + \sum_{j=1}^{m} u_j \nabla g_j(x_i, s_1 \dots\ s_k, s) + & & \\ + \sum_{j=1}^{n} v_j \nabla f_j(x_i, s_1 \dots\ s_k, s) & = & 0 \\ u_j g_j(\mathbf{x}; s_1 \dots\ s_k, s) & = & 0 \\ (j = 1 \dots m) & & \\ f_j(\mathbf{x}; s_1 \dots\ s_k, s) & = & 0 \\ (j = 1 \dots n) & & \\ u_j & \geq & 0 \\ (j = 0 \dots m) & & \\ g_j(\mathbf{x}; s_1 \dots\ s_k, s) & \leq & 0 \\ \sum_{j=0}^{m} u_j + \sum_{j=1}^{n} v_j^2 & = & 1 \end{array} \right.$$

The last condition is a normalization condition which implies that $u_j \in [0, 1]$ (since $u_j \geq 0$) and $v_j \in [-1, 1]$.

Theorem 1 (Projection of geometric primitive)**.** *Let $A : f(\mathbf{x}; s)$ be a geometric primitive. When projecting down k dimensions (eliminating $x_1 \dots x_k$), the projec-*

tion can be specified by:

$$\pi^k(A) \longrightarrow \frac{\partial f}{\partial x_1} = \frac{\partial f}{\partial x_2} = \ldots = \frac{\partial f}{\partial x_k} = 0$$

Proof. Let $L = us + vf(\mathbf{x}; s)$. Then $F_\pi(\mathbf{x^{1\ldots k}}; s) =$

$$\left\{\begin{array}{lcll} u + v\frac{\partial f}{\partial s} & = & 0 & (L'_s) \\ v\frac{\partial f}{\partial x_1} & = & 0 & (L'_{x_1}) \\ & \vdots & & \\ v\frac{\partial f}{\partial x_k} & = & 0 & (L'_{x_k}) \\ u + v^2 & = & 1 & \\ f(\mathbf{x}; s) & = & 0 & \end{array}\right. \Longleftrightarrow$$

$$\left\{\begin{array}{lcl} v^2 - \frac{\partial f}{\partial s}v - 1 & = & 0 \\ \frac{\partial f}{\partial x_1} & = & 0 \\ & \vdots & \\ \frac{\partial f}{\partial x_k} & = & 0 \\ v & \neq & 0 \\ u & = & 1 - v^2 \\ f(\mathbf{x}; s) & = & 0 \end{array}\right.$$

Therefore, to optimize s one has to look at the critical points where each derivative with respect to x_i vanishes, *i.e.*,

$$\frac{\partial f}{\partial x_1} = \frac{\partial f}{\partial x_2} = \cdots = \frac{\partial f}{\partial x_k} = 0.$$

□

Theorem 2 (Projection of dominant set)**.**

$$\pi^k(A|B) \;=\; \pi^k(A)|B \;\cup\; \pi^k(A \bowtie B)$$

Proof. Consider $A : f_0(\mathbf{x}; s), B : f_1(\mathbf{x}; s_1)$. Let $L = u_0 s + u_1(s_1 - s) + v_0 f_0(\mathbf{x}; s) + v_1 f_1(\mathbf{x}; s_1)$. Then $F_\pi(\mathbf{x^{1\ldots k}}; s) =$

$$\left\{\begin{array}{lcll} u_0 - u_1 + v_0\frac{\partial f_0}{\partial s} & = & 0 & (L'_s) \\ u_1 + v_1\frac{\partial f_1}{\partial s_1} & = & 0 & (L'_{s_1}) \\ v_0\frac{\partial f_0}{\partial x_1} + v_1\frac{\partial f_1}{\partial x_1} & = & 0 & (L'_{x_1}) \\ & \vdots & & \\ v_0\frac{\partial f_0}{\partial x_k} + v_1\frac{\partial f_1}{\partial x_k} & = & 0 & (L'_{x_k}) \\ u_0 + u_1 + v_0^2 + v_1^2 & = & 1 & \\ u_1(s_1 - s) & = & 0 & \\ u_j & \geq & 0 & (j = 0, 1) \\ f_0(\mathbf{x}; s) & = & 0 & \\ f_1(\mathbf{x}; s_1) & = & 0 & \end{array}\right.$$

Case $s > s_1$**.** It follows that $u_1 = 0$. Then

$$\left\{\begin{array}{lcl} u_0 + v_0\frac{\partial f_0}{\partial s} & = & 0 \\ v_1\frac{\partial f_1}{\partial s_1} & = & 0 \\ v_0\frac{\partial f_0}{\partial x_1} + v_1\frac{\partial f_1}{\partial x_1} & = & 0 \\ & \vdots & \\ v_0\frac{\partial f_0}{\partial x_k} + v_1\frac{\partial f_1}{\partial x_k} & = & 0 \\ u_0 + v_0^2 + v_1^2 & = & 1 \\ f_0(\mathbf{x}; s) & = & 0 \\ f_1(\mathbf{x}; s_1) & = & 0 \end{array}\right.$$

- $v_1 = 0$.

$$\left\{\begin{array}{lcl} u_0 + v_0\frac{\partial f_0}{\partial s} & = & 0 \\ v_0\frac{\partial f_0}{\partial x_1} & = & 0 \\ & \vdots & \\ v_0\frac{\partial f_0}{\partial x_k} & = & 0 \\ u_0 + v_0^2 & = & 1 \\ f_0(\mathbf{x}; s) & = & 0 \\ f_1(\mathbf{x}; s_1) & = & 0 \end{array}\right.$$

Which implies that $v_0 \neq 0$. In this case, $\frac{\partial f_1}{\partial x_1} = \ldots = \frac{\partial f_1}{\partial x_k} = 0$. Finally

$$\left\{\begin{array}{lcl} u_0 & = & 1 - v_0^2 \\ 0 & = & v_0^2 - v_0\frac{\partial f_0}{\partial s} - 1 \end{array}\right.$$

The discriminant of the quadratic polynomial with respect to v_0 equals $(\frac{\partial f_0}{\partial s})^2 + 4 > 0$ which means there always exists a real solution with respect to v_0. This solution lies in $[-1, 1]$ as required. The interesting constraint we obtained is that

$$\frac{\partial f_1}{\partial x_1} = \frac{\partial f_1}{\partial x_2} = \ldots = \frac{\partial f_1}{\partial x_k} = 0.$$

- $\frac{\partial f_1}{\partial s_1} = 0 \Rightarrow v_1 \in [-1, 1]$ and $u_0 > 0$.

$$\left\{\begin{array}{lcl} u_0 + v_0\frac{\partial f_0}{\partial s} & = & 0 \\ v_0\frac{\partial f_0}{\partial x_1} + v_1\frac{\partial f_1}{\partial x_1} & = & 0 \\ & \vdots & \\ v_0\frac{\partial f_0}{\partial x_k} + v_1\frac{\partial f_1}{\partial x_k} & = & 0 \\ u_0 + v_0^2 + v_1^2 & = & 1 \\ f_0(\mathbf{x}; s) & = & 0 \\ f_1(\mathbf{x}; s_1) & = & 0 \end{array}\right.$$

This solution set contains the previous case when $v_1 = 0$. Nevertheless we solve this system to obtain more general conditions. We set $\chi_{ik} = \frac{\partial f_i}{\partial x_k}$ and

$\sigma_0 = \frac{\partial f_0}{\partial s}$:

$$\begin{cases} \left(v_0 - \frac{\sigma_0}{2}\right)^2 + v_1^2 & = & 1 + \frac{\sigma_0^2}{4} \\ v_0\chi_{01} + v_1\chi_{11} & = & 0 \\ & \vdots & \\ v_0\chi_{0k} + v_1\chi_{1k} & = & 0 \\ u_0 & = & 1 - v_0^2 - v_1^2 \\ f_0(\mathbf{x}; s) & = & 0 \\ f_1(\mathbf{x}; s_1) & = & 0 \end{cases}$$

The solution set with respect to (v_0, v_1) lies at the intersection of a circle centered at $\left(\frac{\sigma_0}{2}, 0\right)$ and k lines passing through the origin, the slope of which is determined by $(\chi_{0j}, \chi_{1j}),\ j = 1 \dots k$. If $k = 1$ then the line intersects the circle in 2 points in general, otherwise the set of lines has to be coincident (since we have a homogeneous system). That is there exist $\binom{k}{2}$ extra constraints which force each pair of lines to be parallel. These can be expressed in terms of the Jacobian determinant: $\forall i, j,\ 1 \leq i < j \leq k$

$$\begin{vmatrix} \chi_{0i} & \chi_{0j} \\ \chi_{1i} & \chi_{1j} \end{vmatrix} = 0 \iff J_{ij}(f_0, f_1) = 0.$$

Case $s = s_1$. We have:

$$\begin{cases} u_0 - u_1 + v_0\frac{\partial f_0}{\partial s} & = & 0 \\ u_1 + v_1\frac{\partial f_1}{\partial s_1} & = & 0 \\ v_0\frac{\partial f_0}{\partial x_1} + v_1\frac{\partial f_1}{\partial x_1} & = & 0 \\ & \vdots & \\ v_0\frac{\partial f_0}{\partial x_k} + v_1\frac{\partial f_1}{\partial x_k} & = & 0 \\ u_0 + u_1 + v_0^2 + v_1^2 & = & 1 \\ u_1 & \geq & 0 \\ f_0(\mathbf{x}; s) & = & 0 \\ f_1(\mathbf{x}; s_1) & = & 0 \end{cases}$$

There are many solutions, but at least one is considered in the previous case, when $u_1 = v_1 = 0$.

A better approach is to consider a different Lagrangian when $s = s_1$ in order to avoid simultaneous vanishing of all constraints. In this case we have $L = u_0 s + v_0 f_0(\mathbf{x}; s) + v_1 f_1(\mathbf{x}; s)$ which leads to:

$$F\pi(\mathbf{x}^{\mathbf{1}\dots\mathbf{k}}; s) = \begin{cases} u_0 + v_0\frac{\partial f_0}{\partial s} + v_1\frac{\partial f_1}{\partial s} & = & 0 \\ v_0\frac{\partial f_0}{\partial x_1} + v_1\frac{\partial f_1}{\partial x_1} & = & 0 \\ & \vdots & \\ v_0\frac{\partial f_0}{\partial x_k} + v_1\frac{\partial f_1}{\partial x_k} & = & 0 \\ u_0 + v_0^2 + v_1^2 & = & 1 \\ u_0 & \geq & 0 \\ f_0(\mathbf{x}; s) & = & 0 \\ f_1(\mathbf{x}; s) & = & 0 \end{cases}$$

We set $\chi_{ik} = \frac{\partial f_i}{\partial x_k}$ and $\sigma_i = \frac{\partial f_i}{\partial s}$:

$$\begin{cases} \left(v_0 - \frac{\sigma_0}{2}\right)^2 + \left(v_1 - \frac{\sigma_1}{2}\right)^2 & = & 1 + \frac{\sigma_0^2}{4} + \frac{\sigma_1^2}{4} \\ v_0\chi_{01} + v_1\chi_{11} & = & 0 \\ & \vdots & \\ v_0\chi_{0k} + v_1\chi_{1k} & = & 0 \\ u_0 & = & 1 - v_0^2 - v_1^2 \\ u_0 & \geq & 0 \\ f_0(\mathbf{x}; s) & = & 0 \\ f_1(\mathbf{x}; s) & = & 0 \end{cases}$$

The solution set with respect to (v_0, v_1) lies at the intersection of a circle centered at $\left(\frac{\sigma_0}{2}, \frac{\sigma_1}{2}\right)$ and k lines passing through the origin, the slope of which is determined by $(\chi_{0j}, \chi_{1j}),\ j = 1 \dots k$. If $k = 1$ then the line intersects the circle in 2 points in general, otherwise the set of lines has to be coincident (since we have a homogeneous system). That is there exist $\binom{k}{2}$ extra constraints which force each pair of lines to be parallel. These can be expressed in terms of the Jacobian determinant: $\forall i, j,\ 1 \leq i < j \leq k$

$$\begin{vmatrix} \chi_{0i} & \chi_{0j} \\ \chi_{1i} & \chi_{1j} \end{vmatrix} = 0 \iff J_{ij}(f_0, f_1) = 0. \quad \square$$

References

[1] Stephen Cameron. Approximation hierarchies and s-bounds. In *Symposium on Solid Modeling and Applications*, pages 129–137, 1991.

[2] Stephen Cameron and Chee-Keng Yap. Refinement methods for geometric bounds in constructive solid geometry. *ACM Trans. Graph.*, 11(1):12–39, 1992.

[3] Gilles Chabert and Luc Jaulin. Contractor programming. *Artificial Intelligence*, 173(11):1079 – 1100, 2009.

[4] N. Delanoue, L. Jaulin, and B. Cottenceau. Guaranteeing the homotopy type of a set defined by nonlinear inequalities. *Reliable computing*, 13(5):381–398, 2007.

[5] N. Delanoue, L. Jaulin, and B. Cottenceau. Using interval arithmetic to prove that a set is path-connected. *Theoretical Computer Science, Special issue: Real Numbers and Computers*, 351(1):119–128, February 2006.

[6] Eldon Hansen. *Global optimization using interval analysis*. Monographs and textbooks in pure and applied mathematics. M. Dekker, New York, 1992.

[7] Aristides A. G. Requicha and Robert B. Tilove. Mathematical foundations of constructive solid geometry: General topology of closed regular sets. Technical report, UR Research [http://dspace.lib.rochester.edu/oai/request] (United States), 1978.

[8] Aristides G. Requicha. Representations for rigid solids: Theory, methods, and systems. *ACM Comput. Surv.*, 12(4):437–464, December 1980.

[9] Vadim Shapiro and Donald L. Vossler. Construction and optimization of CSG representations. *Computer-Aided Design*, 23(11):4 – 20, 1991.

[10] W. A. Stein et al. *Sage Mathematics Software (Version 5.0.1).* The Sage Development Team, 2012. http://www.sagemath.org.

[11] Natee Tongsiri. Constructive Solid Geometry with Projection: An Approach to Piano Movers' Problem, 2006.

[12] George Tzoumas, Jean-Marc Cane, Arnaud Kubicki, Dominique Michelucci, and Sebti Foufou. Une nouvelle représentation pour les ensebles géométriques. In *Journées du Groupe de travail en Modélisation Géométrique*, Marseille, France, 2013.

[13] George Tzoumas and Dominique Michelucci. A new representation for geometric sets and a nontrivial application. Technical report, CNRS UMR 5158, Le2i, University of Burgundy, France, 2012.

[14] Eric W. Weisstein. Determinant. From MathWorld—A Wolfram Web Resource, 2013.

[15] Frank Zizza. Differential forms for constrained max-min problems: Eliminating lagrange multipliers. *The College Mathematics Journal*, 29(5):pp. 387–396, 1998.