Chapter 1

Nef Polyhedra for Boolean Operations

For boolean operations on two polyhedra in highly dimensional space (in d dimensions, where d is 2, 3 or larger) we represent a polyhedron as a "set of pyramids". The computation of the complement and of the closure of a polyhedron, as well as the intersection of two polyhedra can be reduced to the application of primitive operations on pyramids or on sets of pyramids. The union and the difference of two polyhedra P_1 and P_2 can be derived from the complementation and interesection operations on polyhedra:

$$P_1 \bigcup P_2 = cpl(cpl(P_1) \bigcap cpl(P_2)), P_1 P_2 = P_1 \bigcap cpl(P_2).$$

Definition

A set Q in $\mathbb{E}^d$ is called a *cone* if there exists a point $x \in \mathbb{E}^d$ such that $Q = x + \mathbb{E}^+(Q - x)$ (with $\mathbb{E}^+ = \{\lambda \in \mathbb{E}; \lambda > 0\}$). The point x is then called an *apex* of Q. The set of all apices of Q is denoted by N(Q).

Definition

A set Q in $\mathbb{E}^d$ is called a *pyramid* if Q is a polyhedron and a cone.

A pyramid can be represented by a (finite) set of hyperplanes in $\mathbb{E}^d$ and for each hyperplane the information on which side of the hyperplane the pyramid lies.

We now describe the two classes we use for boolean operations on polyhedra: *CGAL_Pyramid_d*, *CGAL_Set_of_pyramids_d*.

A *CGAL_Pyramid_d* is defined by the use of a finite set H of hyperplanes in $\mathbb{E}^d$ and an *arrangement* *C(H)* (also called *cell complex*). Each hyperplane is represented by a $(d+1)$-tuple of coefficients. The arrangement is represented by a set of sign-tuples: let n be the cardinality of set H, for each cell in the arrangement defined by H a tuple of n signs determines on which side of each hyperplane the cell lies. A *CGAL_Pyramid_d Q* will consists of a set H of $(d+1)$-tuples of coefficients and a subset *T(H,Q)* of the set of sign-tuples of the arrangement *C(H)*.

A *CGAL_Set_of_pyramids_d* is a set of *CGAL_Pyramid_d*'s.

Implementation

The general idea is that a polyhedron P is represented by a set of pyramids QP. For a point $x \in \mathbb{E}^d$ there exists a neighborhood U_0 of x such that for every U, $x \in U \subset U_0 \subset P : x + \mathbb{E}^+(U_0 - x) = x + \mathbb{E}^+(U - x)$. Denote P^x for the pyramid $x + \mathbb{E}^+(U_0 - x)$ with point x as an apex. The polyhedron

P determines a finite set of such pyramids P^x. The equivalence classes of the relation x y iff. $P^x = P^y$ are called *faces* of P. We denote with $S(P)$ the set of all faces of P. For a face S we denote P^S for P^x with $x \in S$. We now have: $QP = \{Q = P^S; S \in S(P)\}$. The method is described in [H. Bieri and W. Nef].

Operations

on *CGAL_Pyramid_d*: let Q, Q_1 and Q_2 be *CGAL_Pyramid_d*'s.

... *merge_complex(... T(H_1,Q_1), ... T(H_2,Q_2))*

returns a cell complex compatible with Q_1 and Q_2.

bool *is_equal(... T(H_1,Q_1), ... T(H_2,Q_2))*

returns *true* if $Q_1 = Q_2$.

bool *shrink_set_of_hyperplanes(... T(H,Q), ... H_0)*

precondition: $H_0 \subset H$. returns *true* if H_0 is compatible with $C(H)$.

CGAL_Pyramid_d *complement(CGAL_Pyramid_d Q)*

returns the complement of Q.

CGAL_Pyramid_d *intersect_with_apex_condition(CGAL_Pyramid_d Q_1, CGAL_Pyramid_d Q_2)*

precondition: Q_1 and Q_2 have at least one apex in common. returns the pyramid $(Q_1 \bigcap Q_2)$.

... *apices(CGAL_Pyramid_d Q)*

returns set of apices of Q (that's a set of hyperplanes).

... *intersect_with_theta_hyperplane(CGAL_Pyramid_d Q, NT theta)*

precondition: Q is in general position, $\dim(N(Q)) > 0$, $\dim(N(Q) \bigcap H($-theta$))=\dim(N(Q))$-1. returns $Q \bigcap H($-theta$)$. (Where $H($-theta$)$ is the hyperplane of all points with first coordinate equal to -theta.)

CGAL_Pyramid_d *find_Q(... Set, CGAL_Pyramid_d Q)*

returns pyramid Q for which $Q \bigcap H($-theta$)$ = Set.

CGAL_Pyramid_d *pyramid_of(CGAL_Pyramid_d Q, CGAL_Point x)*

returns Q^x

CGAL_Set_of_pyramids_d *set_of_pyramids_of(CGAL_Pyramid_d Q)*

returns the finite set of all Q^x ($x \in \mathbb{E}^d$).

CGAL_Pyramid_d *closure(CGAL_Pyramid_d Q)*

returns the closure of Q.

Operations

on *CGAL_Set_of_pyramids_d*: let QP denote the set of pyramids representing polyhedron P.
incidence_relation(CGAL_Set_of_pyramids_d QP)

at instantiation time the incidence relation on pyramids in the set $Q(P)$, corresponding to incidence of faces of polyhedron P represented by QP, is constructed.

bool *face_of_P(CGAL_Set_of_pyramids_d QP, ... S)*

returns true if $S \subset P$ where $Q = P^S \in QP, S \in S(P)$ is meant.

bool *point_on_face(CGAL_Set_of_pyramids_d QP,*
... S,
CGAL_Point x)

returns true if $x \in S$, where $Q = P^S \in QP, S \in S(P)$ is meant.

CGAL_Set_of_pyramids_d *unbounded_part(CGAL_Set_of_pyramids_d QP, ... sign)*

(sign can be "plus" or "minus") returns $Q_{+\infty}(P)$ if sign is "plus", returns $Q_{-\infty}(P)$ if sign is "minus", where $Q_{+/-\infty}(P)$ is the subset of QP corresponding to the unbounded faces of P, unbounded to left or right respectively.

Remarks:

1) When we have a polyhedron given in another representation (e.g. for 2 dimensions, as a *CGAL_Polygon*, *CGAL_Triangle*,...) the set of pyramids must be constructed.

2) From pyramid Q, the arrangement has to be computed with a plane sweep as in [H. Edelsbrunner] or [H. Bieri and W. Nef], Chapter 7.

3) (E1), (E2), (E3), (E5) are not purely methods on class *CGAL_Pyramid_d* nor are the other operations in the appropriate format yet.

4) In (A4), sign is some kind of flag, and should be implemented that way.

Problems still:

1) Each hyperplane is represented by a $(d + 1)$-tuple of coefficients. Do we make a "*CGAL_Hyperplane*" (in (E3), (P3)) ?

2) An arrangement (as in (E1), (E2), (E3)) should be represented in some way or another.

3) In (P4) and P(5), what is $Q \bigcap H$(-theta) and *Set* ?

4) What is S in (A2)?

5) Implementation of (A2), (A3), (A4) is unclear.

Evaluation: In our opinion the Nef Polyhedra make an interesting approach to Boolean Operations possible. However, the algorithms are not entirely clear for the moment. Especially the algorithms to transform other representation for polyhedra into the representation of Nef polyhedra are not worked out in a practical way in the literature. Also the preconditions on "general position" for some algorithms need further study. On the short term, and for a basic structure for Boolean Operations on Polyhedra in CGAL we find Nef Polyhedra unsuited. As a research topic in a later stadium we might however revisit the Nef Polyhedra.

Bibliography

[H. Bieri and W. Nef] H. Bieri and W. Nef, Elementary Set Operations with d-Dimensional Polyhedra, in LNCS 555.

[H. Bieri and W. Nef] H. Bieri and W. Nef, A Recursive Sweep-Plane Algorithm, Determining All Cells of a Finite Division of R^d, in Computing, Vol. 28, pp. 189-198, 1982.

[H. Bieri and W. Nef] H. Bieri and W. Nef, A Sweep-Plane Algorithm for Computing the Euler-Characteristic of Polyhedra Represented in Boolean Form, in Computing Vol. 34, No. 4, pp. 287-302, 1985.

[H. Bieri and W. Nef] H. Bieri and W. Nef, A Sweep-Plane Algorithm for Computing the Volume of Polyhedra Represented in Boolean Form, in Linear Algebra and its Applications, No. 52/53, pp. 69-97, 1983.

[H. Edelsbrunner] H. Edelsbrunner, Algorithms in Combinatorial Geometry, Springer Verlag.

[V. Ferrucci and A. Paoluzzi] V. Ferrucci and A. Paoluzzi, Extrusion and Boundary Evaluation for Multidimensional Polyhedra, in Computer Aided Design, Vol. 23, Nr. 1, pp. 40-50, 1991.

[P. Lienhardt] P. Lienhardt, Topological Models for Boundary Representation - A Comparison with N-Dimensional Generalized Maps, in Computer Aided Design, Vol. 23, Nr. 1, pp. 59-82, 1991.

[JR. Rossignac and AA. Requicha] JR. Rossignac and AA. Requicha, Constructive Non-Regularized Geometry, in Computer Aided Design, Vol. 23, Nr. 1, pp. 21-32, 1991.