Minkowski Sum of Convex Polyhedra

Efi Fogel

Tel Aviv University, Israel

Applied Aspects of Computational Geometry
Tel Aviv University, April 2009

Minkowski Sum Definition

Definition (Minkowski sum)

Let P and Q be two point sets in $\mathbb{R}^d$. The Minkowski sum of P and Q, denoted as $P \oplus Q$, is the point set $\{p + q \,|\, p \in P, q \in Q\}$.

- Applies to every dimension d.
  - Today we concentrate at the case $d = 3$.
- Applies to arbitrary point sets.

Polytope Definition

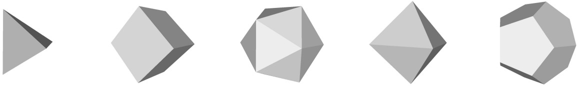
Definition (convex polyhedron)

A convex set $Q \subseteq \mathbb{R}^d$ given as an intersection of finite number of closed half-spaces $H = \{h \in \mathbb{R}^d \,|\, Ah \leq B\}$ is called convex polyhedron.

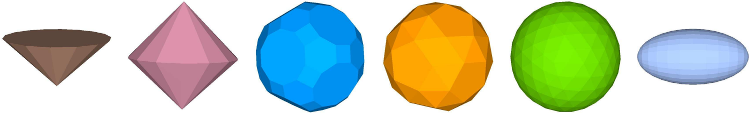
Definition (polytope)

A bounded convex polyhedron $P \subset \mathbb{R}^d$ is called polytope.

The 5 Platonic polytopes:



tetrahedron, cube, icosahedron, octahedron, dodecahedron



dioctagonal pyramid, dioctagonal dipyramid, truncated icosi-dodecahedron, pentagonal hexecontahe-dron, geodesic sphere level 4, ellipsoid

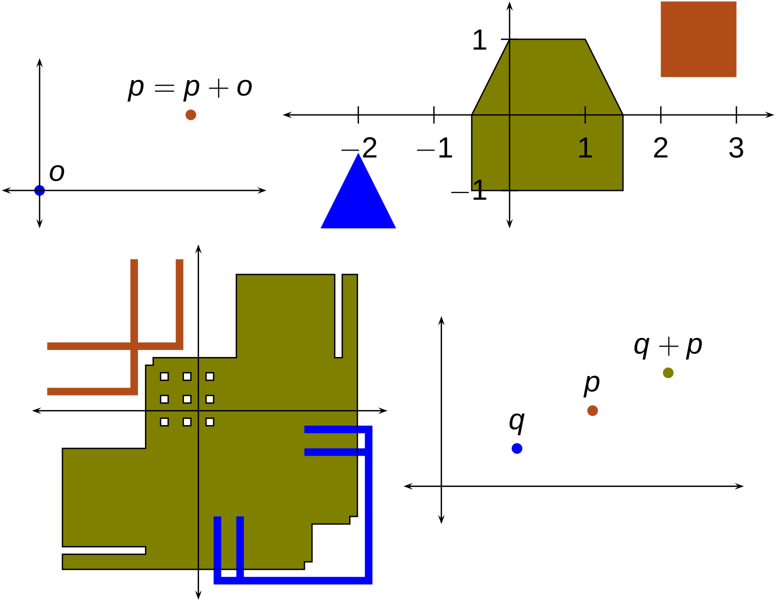
Hyperplanes

Definition (supporting hyperplane)

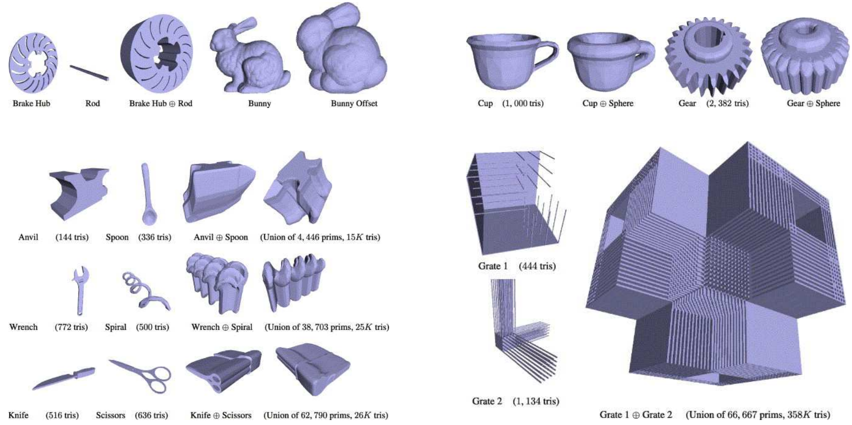
A hyperplane h supports a set $P \subset \mathbb{R}^d$ (at c) if P intersects h (at c) and is contained in one of the closed halfspaces bounded by h.

- If p is a boundary point of a polytope P, then there exists a supporting hyperplane at p.
  - If p is contained in a facet, there exists a single supporting hyperplane at p.
  - If p lies in an edge or coincides with a vertex, there are many supporting hyperplane at p.

Minkowski Sum Examples in $\mathbb{R}^2$



Minkowski Sum Examples in $\mathbb{R}^3$



[1]

[1] All figures taken from [VM06].

Minkowski Sum Properties

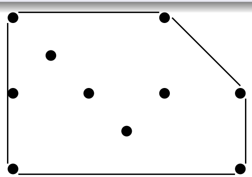
- The Minkowski sum of two (non-parallel) line segments in $\mathbb{R}^2$ is a convex polygon.
- The Minkowski sum of two (non-parallel) polygons in $\mathbb{R}^3$ is a convex polyhedron.
- $P = P \oplus \{o\}$, where o is the origin.
- If P and Q are convex, then $P \oplus Q$ is convex.
- $P \oplus Q = Q \oplus P$.
- $\lambda(P \oplus Q) = \lambda P \oplus \lambda Q$, where $\lambda P = \{\lambda p \,|\, p \in P\}$.
- $2P \subseteq P \oplus P$, $3P \subseteq P \oplus P \oplus P$, etc.
- $P \oplus (Q \cup R) = (P \oplus Q) \cup (P \oplus R)$.

Convex Hull

Definition (convex hull)

The convex hull of a set of points $P \subseteq \mathbb{R}^d$, denoted as $\mathrm{conv}(P)$, is the smallest (inclusionwise) convex set containing P.

When an elastic band stretched open to encompass the input points is released, it assumes the shape of the convex hull.



n — the number of input points.
h — the number of points in the hull.

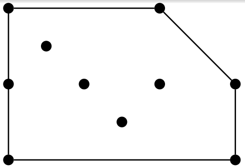
- Time complexities of convex hull computation:
  - Optimal, output sensitive: $O(n \log h)$. [Chan06]
  - QuickHull (expected): $O(n \log n)$. [BDH96]

Convex Hull

Definition (convex hull)

The convex hull of a set of points $P \subseteq \mathbb{R}^d$, denoted as $\mathrm{conv}(P)$, is the smallest (inclusionwise) convex set containing P.

When an elastic band stretched open to encompass the input points is released, it assumes the shape of the convex hull.



n — the number of input points.
h — the number of points in the hull.

- Time complexities of convex hull computation:
  - Optimal, output sensitive: $O(n \log h)$. [Chan06]
  - QuickHull (expected): $O(n \log n)$. [BDH96]

Minkowski-Sum Construction: Convex Hull

Observation

The Minkowski sum of two polytopes P and Q is the convex hull of the pairwise sums of vertices of P and Q, respectively.

```
typedef CGAL::Exact_predicates_exact_constructions_kernel  Kernel;
typedef Kernel::Point_3                                    Point;
typedef Kernel::Vector_3                                   Vector;
typedef CGAL::Polyhedron_3<Kernel>                         Polyhedron;

  std::vector<Point> in1, in2, points;
  // Process input ...
  points.resize(in1.size() * in2.size());
  std::vector<Point>::const_iterator it1, it2;
  std::vector<Point>::iterator it3 = points.begin();
  for (it1 = in1.begin(); it1 != in1.end(); ++it1) {
    Vector v(CGAL::ORIGIN, *it1);
    for (it2 = in2.begin(); it2 != in2.end(); ++it2) *it3++ = (*it2) + v;
  }
  Polyhedron polyhedron;
  CGAL::convex_hull_3(points.begin(), points.end(), polyhedron);
```

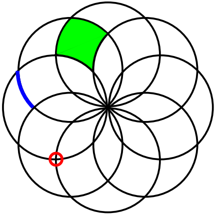
- `CGAL::convex_hull_3` implements QuickHull.
- Time complexities of Minkowski-sum constr. using convex hull:
  - Using `CGAL::convex_hull_3` (expected): $O(nm \log mn)$.
  - Optimal: $O(nm \log h)$.

▸ code

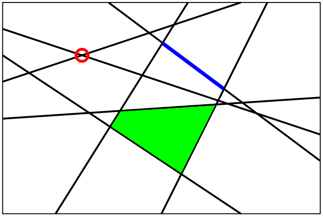
Arrangements on Surfaces in $\mathbb{R}^3$

Definition (arrangement)

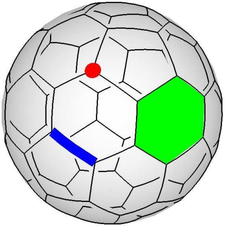
Given a collection $\mathcal{C}$ of curves on a surface, the arrangement $\mathcal{A}(\mathcal{C})$ is the partition of the surface into vertices, edges and faces induced by the curves of $\mathcal{C}$.



An arrangement of circles in the plane



An arrangement of lines in the plane



An arrangement of great-circle arcs on a sphere

Map Overlay

Definition (map overlay)

The map overlay of two planar subdivisions $\mathcal{S}_1$ and $\mathcal{S}_2$, denoted as $\text{overlay}(\mathcal{S}_1, \mathcal{S}_1)$, is a planar subdivision $\mathcal{S}$, such that there is a face f in $\mathcal{S}$ if and only if there are faces f_1 and f_2 in $\mathcal{S}_1$ and $\mathcal{S}_2$ respectively, such that f is a maximal connected subset of $f_1 \cap f_2$.

The overlay of two subdivisions embedded on a surface in $\mathbb{R}^3$ is defined similarly.

n_1, n_2, n — number of vertices in $\mathcal{S}_1$, $\mathcal{S}_2$, $\text{overlay}(\mathcal{S}_1, \mathcal{S}_2)$.

- Time complexities of the computation of the overlay of 2 subdivisions embedded on surfaces in $\mathbb{R}^3$:
  - Using sweep-line: $O((n)\log(n_1 + n_2))$. [BO79]
  - Using trapezoidal decomposition: $O(n)$. [FH95]
    - ★ Precondition: $\mathcal{S}_1$ and $\mathcal{S}_2$ are simply connected.

Gasusian Map of Polytopes

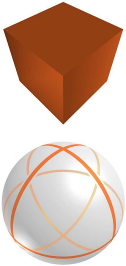
Definition (Gasusian map or normal diagram)

The Gaussian map of a polytope P is the decomposition of $\mathbb{S}^2$ into maximal connected regions so that the extremal point of P is the same for all directions within one region.

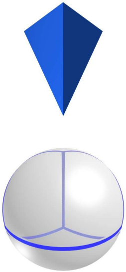
G is a set-valued function from ∂P to $\mathbb{S}^2$.
$G(p \in \partial P) =$ the set of outward unit normals to support planes to P at p.

v, e, f — a vertex, an edge, a facet of P.

- $G(f) =$ outward unit normal to f.
- $G(e) =$ geodesic segment.
- $G(v) =$ spherical polygon.



Cube



tetrahedron

Gasusian Map of Polytopes (cont.)

- $G(P)$ is an arrangement embedded on $\mathbb{S}^2$, where
  - each face $G(v)$ of the arrangement is extended with v.
- $G(P)$ is unique $\Rightarrow G^{-1}(G(P)) = P$.

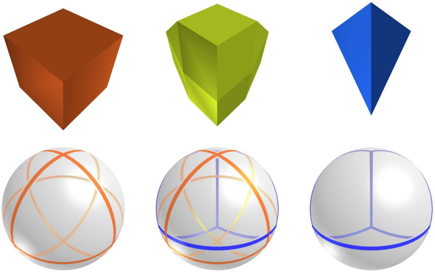
Minkowski-Sums Construction: Gaussian Map

Observation

The overlay of the Gaussian maps of two polytopes P and Q is the Gaussian map of the Minkowski sum of P and Q.

$$\text{overlay}(G(P), G(Q)) = G(P \oplus Q)$$

- The overlay identifies all the pairs of features of P and Q respectively that have common supporting planes.
- These common features occupy the same space on $\mathbb{S}^2$.
- They identify the paiwise features that contribute to $\partial(P \oplus Q)$.



Cube　　Minkowski sum　　tetrahedron

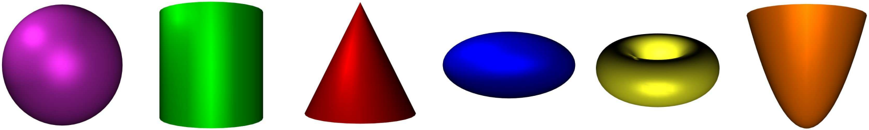
Parametric Surfaces in $\mathbb{R}^3$

Definition (parametric surface)

A parametric surface S of two parameters is a surface defined by parametric equations involving two parameters u and v:

$$f_S(u,v) = (x(u,v),\, y(u,v),\, z(u,v))$$

Thus, $f_S : \mathbb{P} \longrightarrow \mathbb{R}^3$ and $S = f_S(\mathbb{P})$, where $\mathbb{P}$ is a continuous and simply connected two-dimensional parameter space



- We deal with orientable parametric surfaces

The CGAL `Arrangement_on_surface_2` Package

- Constructs, maintains, modifies, traverses, queries, and presents arrangements on two-dimensional parametric surfaces in $\mathbb{R}^3$.
- Robust and exact
  - All inputs are handled correctly (including degenerate input).
  - Exact number types are used to achieve exact results.
- Generic – easy to interface, extend, and adapt.
- Modular – geometric and topological aspects are separated.
- Supports among the others:
  - various point location strategies.
  - zone-construction paradigm.
  - sweep-line paradigm.
  - overlay computation.
- Part of the CGAL basic library. [WFZH08]

Minkowski-Sums Construction: Gaussian Map

m, n, k — number of facets in $P, Q, P \oplus Q$.

- Overlay of CGAL is based on sweep-line.
- $G(P)$ is a simply connected convex subdivision.
- Time complexities of Minkowski-sum constr. using Gaussian map:
  - Using `CGAL::overlay`: $O(k \log(m + n))$.
  - Optimal: $O(k)$.

Map Overlay of CGAL

```
template <class GeomTraitsRed,
          class GeomTraitsBlue,
          class GeomTraitsRes,
          class TopTraitsRed,
          class TopTraitsBlue,
          class TopTraitsRes,
          class OverlayTraits>
void overlay (const Arrangement_on_surface_2<GeomTraitsRed, TopTraitsRed> & arr1,
              const Arrangement_on_surface_2<GeomTraitsBlue, TopTraitsBlue> & arr2,
              Arrangement_on_surface_2<GeomTraitsRes, TopTraitsRes> & arr_res,
              OverlayTraits & ovl_tr)
```

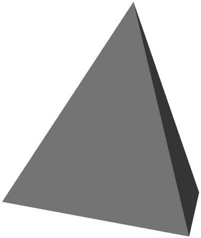
The concept *OverlayTraits* requires the provision of ten functions that handle all possible cases as follows:

1. A new vertex v is induced by coinciding vertices v_r and v_b.
2. A new vertex v is induced by a vertex v_r that lies on an edge e_b.
3. An analogous case of a vertex v_b that lies on an edge e_r.
4. A new vertex v is induced by a vertex v_r that is contained in a face f_b.
5. An analogous case of a vertex v_b contained in a face f_r.
6. A new vertex v is induced by the intersection of two edges e_r and e_b.
7. A new edge e is induced by the overlap of two edges e_r and e_b.
8. A new edge e is induced by the an edge e_r that is contained in a face f_b.
9. An analogous case of an edge e_b contained in a face f_r.
10. A new face f is induced by the overlap of two faces f_r and f_b.

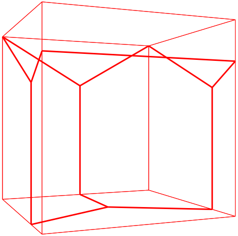
The Cubical Gaussian Map

The Cubical Gaussian Map (CGM) C of a polytope $P \subset \mathbb{R}^3$ is a set-valued function from ∂P to the six faces of the unit cube whose edges are parallel to the major axes and are of length two.

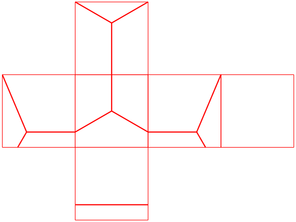
A Tetrahedron



The primal



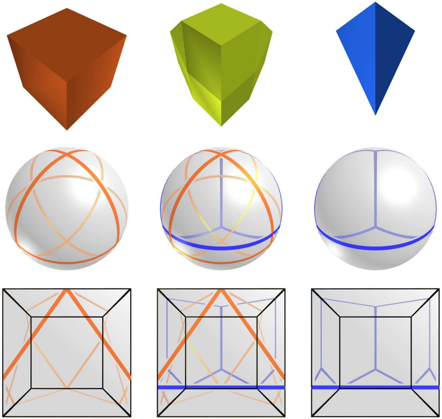
The CGM



The CGM unfolded

Minkowski-Sums Construction: Cubical Gaussian Map

The six overlays of the six pairs of the planar maps of the two cubical Gaussian maps of two polytopes P and Q stiched properly comprise the cubical Gaussian map of the Minkowski sum of P and Q.



Cube Minkowski sum tetrahedron

Minkowski-Sum Construction: Results

Time consumption (in seconds) of the Minkowski-sum computation.

CH — the convex-hull method.

SGM — the (spherical) Gaussian map based method. [BFH+09a]

CGM — the cubical Gaussian-map based method. [FH07]

NGM — the Nef based method. [HKM07]

Fuk — Fukuda's linear-programming based algorithm. [Fuk04]

$\frac{F_1 F_2}{F}$ — the ratio between the product of the number of input facets and the number of output facets.

Summand 1	Summand 2	SGM	CGM	NGM	Fuk	CH	$\frac{F_1 F_2}{F}$
Icosahedron	Icosahedron	0.01	0.01	0.12	0.01	0.01	20.0
DP	ODP	0.04	0.02	0.33	0.35	0.05	2.2
PH	TI	0.13	0.03	0.84	1.55	0.20	10.9
GS4	RGS4	0.71	0.12	6.81	5.80	1.89	163.3
El16	OEl16	1.01	0.14	7.06	13.04	6.91	161.3

DP — dioctagonal pyramid.
PH — pentagonal hexecontahedron.
GS4 — geodesic sphere level 4.
El16 — ellipsoid.

ODP — orthogonal dioctagonal pyramid.
TI — truncated icosidodecahedron.
RGS4 — rotated geodesic sphere level 4.
OEl16 — orthogonal ellipsoid.

Minkowski Sum Application: Collision Detection

- P and Q are two polytopes in $\mathbb{R}^d$.

$P \cap Q \neq \emptyset$ collision detection

Minkowski Sum Application: Collision Detection

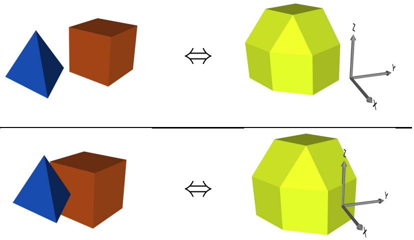
- P and Q are two polytopes in $\mathbb{R}^d$.
- P translated by a vector t is denoted by P^t.

$P \cap Q \neq \emptyset$	collision detection
$\pi(P,Q) = \min\{\|t\| \mid P^t \cap Q \neq \emptyset, t \in \mathbb{R}^d\}$	separation distance
$\delta(P,Q) = \inf\{\|t\| \mid P^t \cap Q = \emptyset, t \in \mathbb{R}^d\}$	penetration depth
$\delta_v(P,Q) = \inf\{\alpha \mid P^{\alpha\vec{v}} \cap Q = \emptyset, \alpha \in \mathbb{R}\}$	directional penetration-depth

Minkowski Sum Application: Collision Detection

- P and Q are two polytopes in $\mathbb{R}^d$.
- P translated by a vector t is denoted by P^t.

$P \cap Q \neq \emptyset \Leftrightarrow \text{Origin} \in M = P \oplus (-Q)$	collision detection
$\pi(P, Q) = \min\{\|t\| \mid P^t \cap Q \neq \emptyset, t \in \mathbb{R}^d\}$	separation distance
$\delta(P, Q) = \inf\{\|t\| \mid P^t \cap Q = \emptyset, t \in \mathbb{R}^d\}$	penetration depth
$\delta_v(P, Q) = \inf\{\alpha \mid P^{\alpha\vec{v}} \cap Q = \emptyset, \alpha \in \mathbb{R}\}$	directional penetration-depth



Minkowski Sum Application: Collision Detection

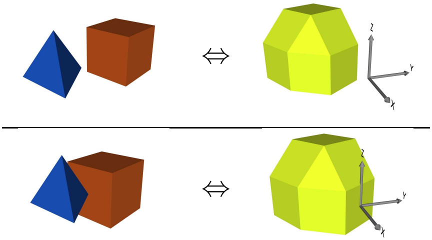
- P and Q are two polytopes in $\mathbb{R}^d$.
- P translated by a vector t is denoted by P^t.

$$P^u \cap Q^w \neq \emptyset \Leftrightarrow w - u \in M = P \oplus (-Q)$$ collision detection

$$\pi(P, Q) = \min\{\|t\| \mid P^t \cap Q \neq \emptyset, t \in \mathbb{R}^d\}$$ separation distance

$$\delta(P, Q) = \inf\{\|t\| \mid P^t \cap Q = \emptyset, t \in \mathbb{R}^d\}$$ penetration depth

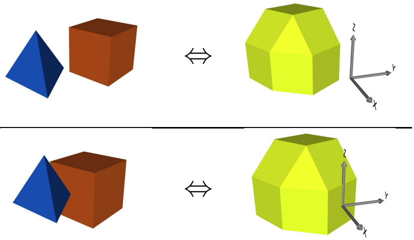
$$\delta_v(P, Q) = \inf\{\alpha \mid P^{\alpha\vec{v}} \cap Q = \emptyset, \alpha \in \mathbb{R}\}$$ directional penetration-depth



Minkowski Sum Application: Collision Detection

- P and Q are two polytopes in $\mathbb{R}^d$.
- P translated by a vector t is denoted by P^t.

$P^u \cap Q^w \neq \emptyset \Leftrightarrow w - u \in M = P \oplus (-Q)$	collision detection
$\pi(P,Q) = \min\{\lVert t \rVert \mid t \in M, t \in \mathbb{R}^d\}$	separation distance
$\delta(P,Q) = \inf\{\lVert t \rVert \mid P^t \cap Q = \emptyset, t \in \mathbb{R}^d\}$	penetration depth
$\delta_v(P,Q) = \inf\{\alpha \mid P^{\alpha\vec{v}} \cap Q = \emptyset, \alpha \in \mathbb{R}\}$	directional penetration-depth



Minkowski Sum Application: Collision Detection

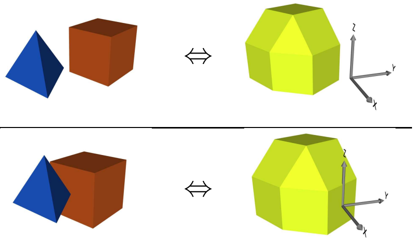
- P and Q are two polytopes in $\mathbb{R}^d$.
- P translated by a vector t is denoted by P^t.

$$P^u \cap Q^w \neq \emptyset \Leftrightarrow w - u \in M = P \oplus (-Q)$$ collision detection

$$\pi(P, Q) = \min\{\|t\| \mid t \in M, t \in \mathbb{R}^d\}$$ separation distance

$$\delta(P, Q) = \inf\{\|t\| \mid t \notin M, t \in \mathbb{R}^d\}$$ penetration depth

$$\delta_v(P, Q) = \inf\{\alpha \mid P^{\alpha\vec{v}} \cap Q = \emptyset, \alpha \in \mathbb{R}\}$$ directional penetration-depth



Minkowski Sum Application: Collision Detection

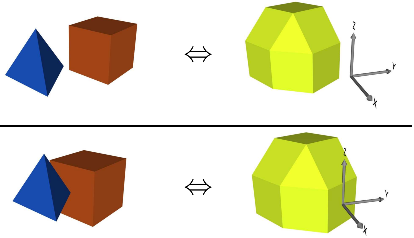
- P and Q are two polytopes in $\mathbb{R}^d$.
- P translated by a vector t is denoted by P^t.

$$P^u \cap Q^w \neq \emptyset \Leftrightarrow w - u \in M = P \oplus (-Q)$$ collision detection

$$\pi(P,Q) = \min\{\|t\| \mid t \in M, t \in \mathbb{R}^d\}$$ separation distance

$$\delta(P,Q) = \inf\{\|t\| \mid t \notin M, t \in \mathbb{R}^d\}$$ penetration depth

$$\delta_v(P,Q) = \inf\{\alpha \mid \alpha\vec{v} \notin M, \alpha \in \mathbb{R}\}$$ directional penetration-depth



Minkowski Sum Application: Collision Detection

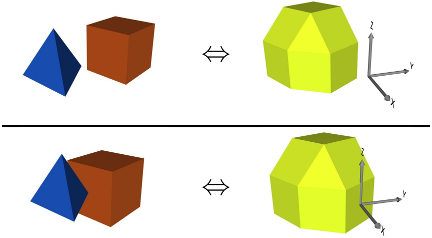
- P and Q are two polytopes in $\mathbb{R}^d$.
- P translated by a vector t is denoted by P^t.

$$P^u \cap Q^w \neq \emptyset \Leftrightarrow w - u \in M = P \oplus (-Q) \qquad \text{collision detection}$$

$$\pi(P^u, Q^w) = \min\{\|t\| \mid (w - u + t) \in M, t \in \mathbb{R}^d\} \qquad \text{separation distance}$$

$$\delta(P^u, Q^w) = \inf\{\|t\| \mid (w - u + t) \notin M, t \in \mathbb{R}^d\} \qquad \text{penetration depth}$$

$$\delta_v(P^u, Q^w) = \inf\{\alpha \mid (w - u + \alpha\vec{v}) \notin M, \alpha \in \mathbb{R}\} \qquad \text{directional penetration-depth}$$



Minkowski Sum Application: Width

Definition (point-set width)

The width of a set of points $P \subseteq \mathbb{R}^d$, denoted as width(P), is the minimum distance between parallel hyperplanes supporting conv(P).

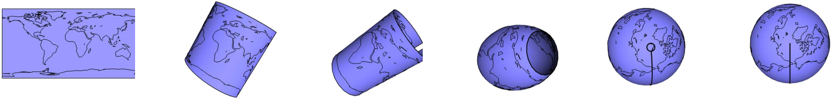
Definition (directional point-set width)

Given a normalized vector v, the directional width, denoted as width$_v(P)$ is the distance between parallel hyperplanes supporting conv(P) and orthogonal to v.

- $width(P) = \delta(P, P) = \inf\{\|t\| \,|\, t \notin (P \oplus -P), t \in \mathbb{R}^d\}$
- Time complexities of width computation in $\mathbb{R}^3$:
  - Applied computation using CGAL Minkowski sum: $O(k \log n)$.
  - Optimal computation using Minkowski sum: $O(k)$.
  - CGAL::Width_3: $O(n^2)$. [FGHHS08]
  - Width optimal computation complexity: subquadratic.

Movies

- Exact Minkowski sums of convex polyhedra.
  - Was presented at the 21st ACM Symposium on Computational Geometry, 2005.
- Arrangements of Geodesic Arcs on the Sphere
  - Was presented at the 24th ACM Symposium on Computational Geometry, 2008.



Related Work

C. Bradford Barber, David P. Dobkin, and Hannu T. Huhdanpaa
The Quickhull algorithm for convex hulls.
ACM Transactions on Mathematical Software, 22(4):469-483, 1996.

Jon Louis Bentley and Thomas Ottmann.
Algorithms for Reporting and Counting Geometric Intersections.
IEEE Transactions on Computers, 28(9): 643–647, 1979.

Eric Berberich, Efi Fogel, Dan Halperin, Michael Kerber, and Ophir Setter.
Arrangements on parametric surfaces ii: Concretizations and applications, 2009.
Manuscript.

Timothy M. Chan
Optimal output-sensitive convex hull algorithms in two and three dimensions.
Discrete & Computational Geometry, 16:361–368, 1996.

Related Work

- Ulrich Finke and Klaus H. Hinrichs.
  Overlaying simply connected planar subdivisions in linear time.
  In *Proceedings of 11th Annual ACM Symposium on Computational Geometry (SoCG)*, pages 119–126. Association for Computing Machinery (ACM) Press, 1995.
- Efi Fogel and Dan Halperin.
  Movie: Exact Minkowski sums of convex polyhedra.
  In *Proceedings of 21st Annual ACM Symposium on Computational Geometry (SoCG)*, pages 382–383. Association for Computing Machinery (ACM) Press, 2005.
- Efi Fogel and Dan Halperin.
  Exact and efficient construction of Minkowski sums of convex polyhedra with applications.
  Computer-Aided Design, 39(11):929–940, 2007.
- Efi Fogel, Dan Halperin, and Christophe Weibel.
  On the exact maximum complexity of Minkowski sums of convex polyhedra.
  Discrete & Computational Geometry.
  Accepted for publication, 2009.

Related Work

Efi Fogel, Ophir Setter, and Dan Halperin.
Movie: Arrangements of geodesic arcs on the sphere.
In *Proceedings of 24th Annual ACM Symposium on Computational Geometry (SoCG)*, pages 218–219. Association for Computing Machinery (ACM) Press, 2008.

Komei Fukuda.
From the zonotope construction to the Minkowski addition of convex polytopes.
Journal of Symbolic Computation, 38(4):1261–1272, 2004.

Peter Hachenberger, Lutz Kettner, and Kurt Mehlhorn.
Boolean operations on 3D selective Nef complexes: Data structure, algorithms, optimized implementation and experiments.
Computational Geometry: Theory and Applications, 38(1-2):64–99, 2007.
Special issue on CGAL.

Gokul Varadhan and Dinesh Manocha.
Accurate Minkowski sum approximation of polyhedral models.
Graphical Models and Image Processing, 68(4):343–355, 2006.

Related Work

Ron Wein, Efi Fogel, Baruch Zukerman, Dan Halperin.

2D Arrangements.

In CGAL Editorial Board, editor, CGAL *User and Reference Manual*. 3.4 edition, 2008.

Kaspar Fischer, Bernd Gärtner, Thomas Herrmann, Michael Hoffmann, and Sven Schönherr.

Optimal Distances.

In CGAL Editorial Board, editor, CGAL *User and Reference Manual*. 3.4 edition, 2008.