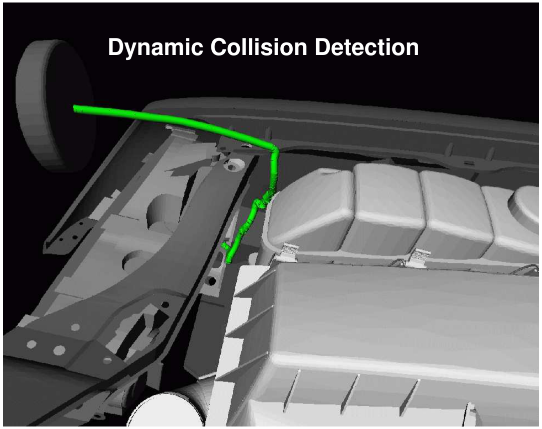
Efficient Distance Computation for Quadratic Curves and Surfaces

Geometric Modeling and Processing 2002

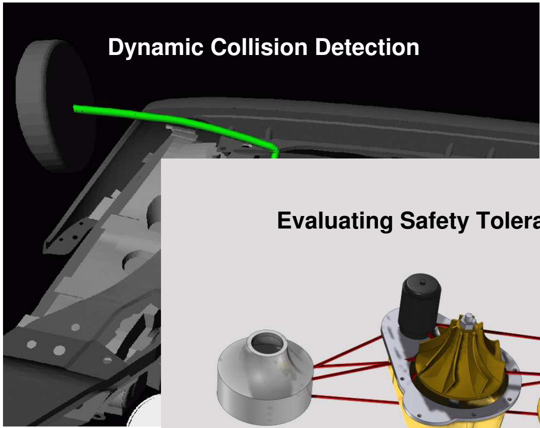
Christian Lennerz and Elmar Schoemer

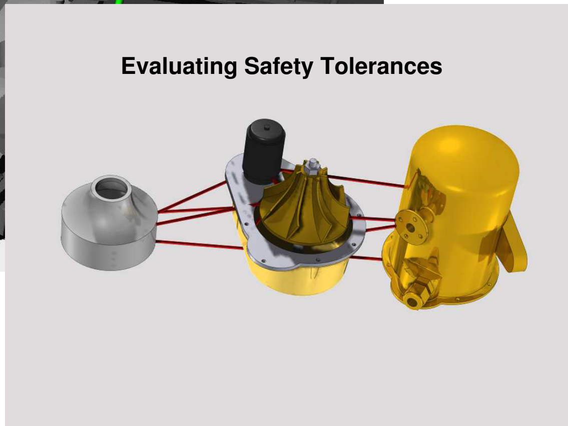
July 10, 2002

Applications

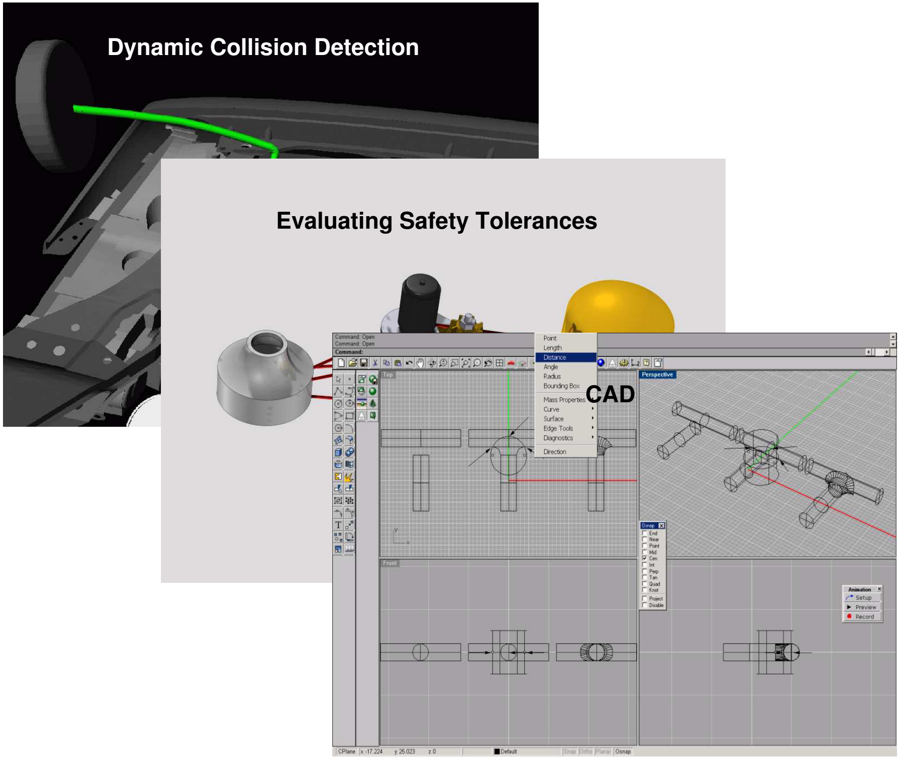


Applications





Applications



Previous Work

- Polyhedral Objects:
  - [*Gilbert,Johnson,Keerthi88*] (GJK)
  - [*Cohen,Lin,Manocha,Ponamgi95*] (I-Collide)
  - [*Cameron97*] (Enhanced GJK)
  - [*Mirtich97*] (V-Clip)
  - [*Larsen,Gottschalk,Lin,Manocha99*] (PQP)
  - [*Kawachi,Suzuki00*]
  - [*Ehmann,Lin01*] (Swift++)

Previous Work

- Polyhedral Objects:
  - [*Gilbert,Johnson,Keerthi88*] (GJK)
  - [*Cohen,Lin,Manocha,Ponamgi95*] (I-Collide)
  - [*Cameron97*] (Enhanced GJK)
  - [*Mirtich97*] (V-Clip)
  - [*Larsen,Gottschalk,Lin,Manocha99*] (PQP)
  - [*Kawachi,Suzuki00*]
  - [*Ehmann,Lin01*] (Swift++)
- Curved Objects:
  - [*Zhou,Sherbrooke,Patrikalakis93*]
  - [*Limaiem,Trochu95*]
  - [*Johnson,Cohen98*] (LUB-Tree)
  - [*Turnbull,Cameron89*]
  - [*Thomas,Turnbull,Ros,Cameron00*]

Conics, Quadrics and Quadratic Complexes

- Quadratic Complexes are polyhedra with faces embedded on quadrics and conics as edges.

Conics, Quadrics and Quadratic Complexes

- Quadratic Complexes are polyhedra with faces embedded on quadrics and conics as edges.
- A quadric is given by an algebraic equation of degree 2:

$$\{\mathbf{x} \in \mathbb{R}^3 \mid \mathbf{x}^\mathsf{T}\mathbf{A}\mathbf{x} + 2\mathbf{a}^\mathsf{T}\mathbf{x} + a_0 = 0\},$$

for a vector $\mathbf{a} \in \mathbb{R}^3$ and symmetric matrix $\mathbf{A} \in \mathbb{R}^{3\times3}$.

Conics, Quadrics and Quadratic Complexes

- Quadratic Complexes are polyhedra with faces embedded on quadrics and conics as edges.
- A quadric is given by an algebraic equation of degree 2:

$$\{\mathbf{x} \in \mathbb{R}^3 \,|\, \mathbf{x}^T \mathbf{A} \mathbf{x} + 2\mathbf{a}^T \mathbf{x} + a_0 = 0\},$$

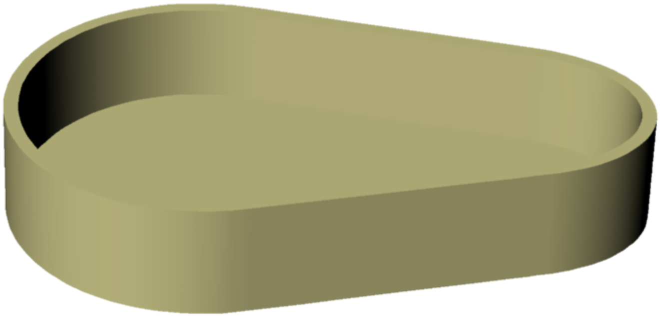
  for a vector $\mathbf{a} \in \mathbb{R}^3$ and symmetric matrix $\mathbf{A} \in \mathbb{R}^{3\times 3}$.

- A conic is explicitly given as the following point set:

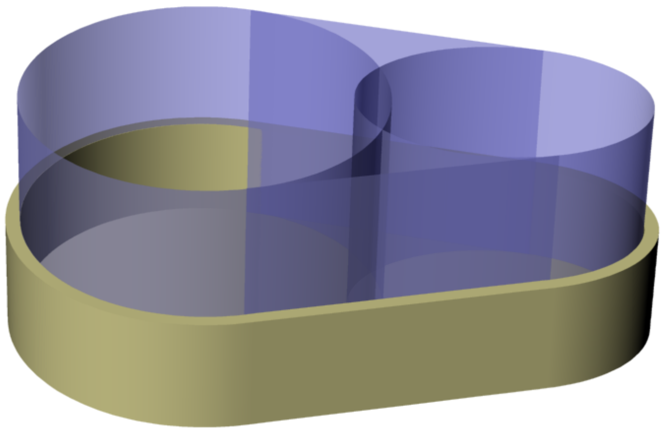
$$\{\mathbf{p} \in \mathbb{R}^3 \,|\, \mathbf{p} \;=\; \mathbf{c} + r(t)\mathbf{u} + s(t)\mathbf{v}\},$$

  where $(r, s) \in \{(\cos, \sin), (\cosh, \sinh), (\mathrm{id}, \mathrm{id}^2), (\mathrm{id}, 0)\}$ and $\mathbf{u}, \mathbf{v} \in \mathbb{R}^3$ with $\mathbf{u}^T \mathbf{v} = 0$.

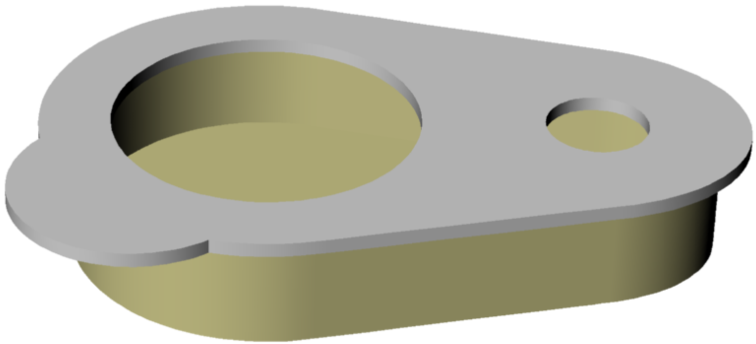
Example of a Quadratic Complex



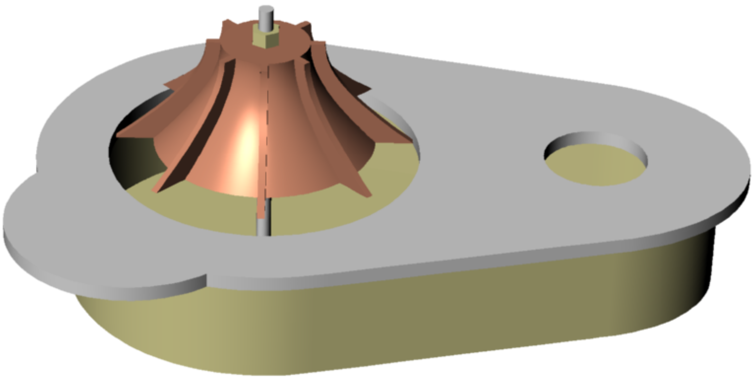
Example of a Quadratic Complex



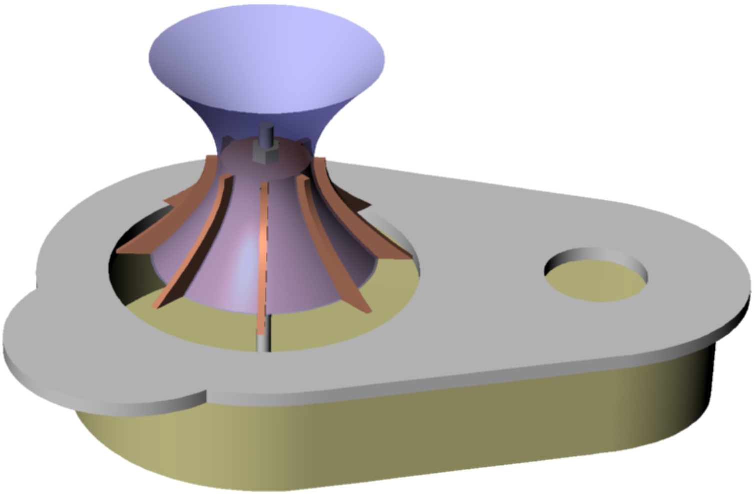
Example of a Quadratic Complex



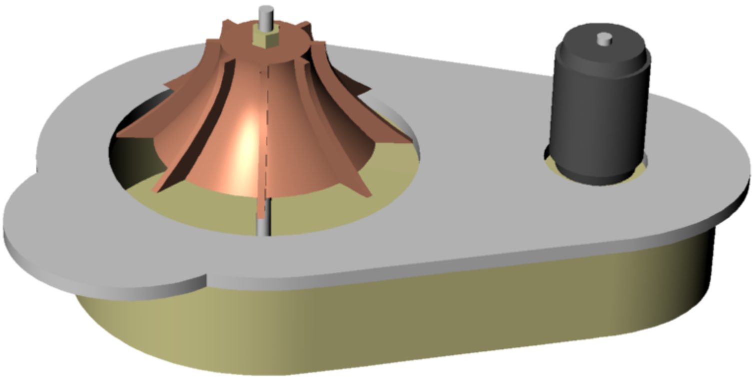
Example of a Quadratic Complex



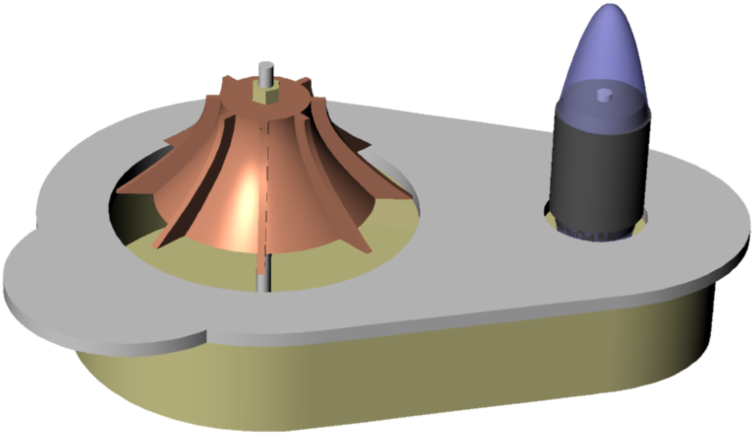
Example of a Quadratic Complex



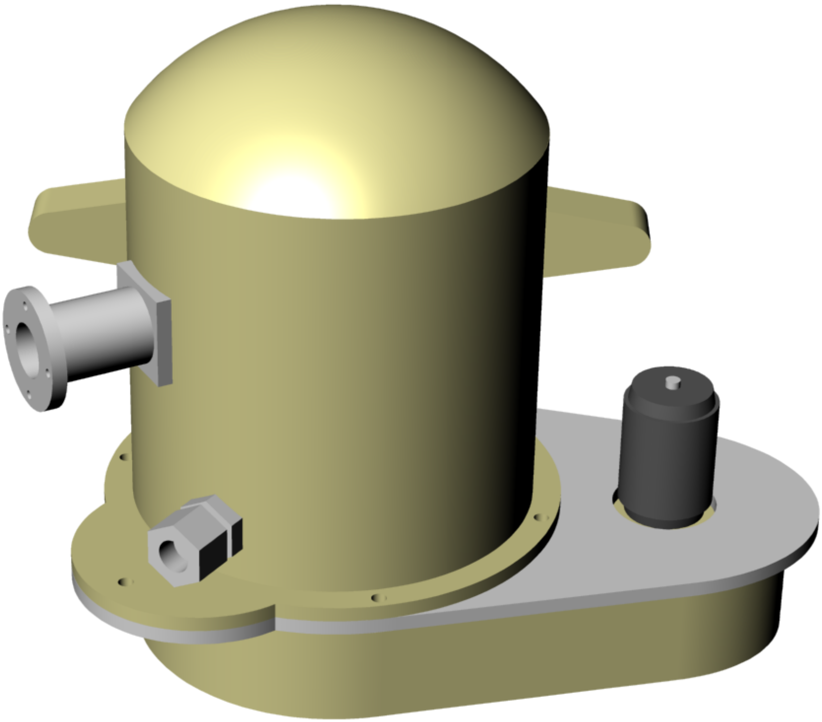
Example of a Quadratic Complex



Example of a Quadratic Complex

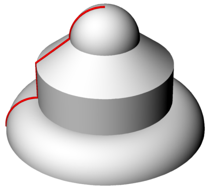


Example of a Quadratic Complex

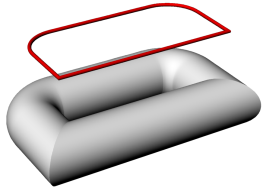


Some Limitations

- Typical CAD-operations on circular profile curves lead to torus patches:



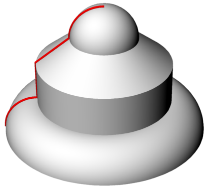
Revolving



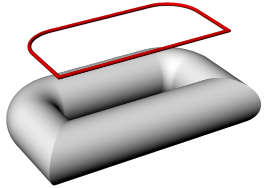
Tubing

Some Limitations

- Typical CAD-operations on circular profile curves lead to torus patches:

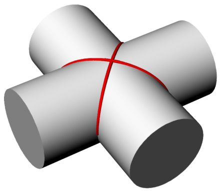


Revolving

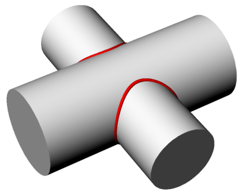


Tubing

- The class of quadratic complexes is not closed under BOOLEAN-operations:



Union (same radii)



Union (different radii)

The Distance Computation Problem

Definition 1 (Distance Computation Problem)
Given two quadratic complexes $\mathbf{C}_1$*,* $\mathbf{C}_2$*. The distance computation problem is to determine the global minimum of the distance function* δ *between the respective point sets, together with a pair of witness points i.e.*

(i) the value $\delta^* := \delta(\mathbf{C}_1, \mathbf{C}_2)$*,*

(ii) a pair of points $(\mathbf{p}, \mathbf{q})$*, s.t.* $\delta^* = \delta(\mathbf{p}, \mathbf{q})$*,*

where δ *denotes the (quadratic)* EUCLIDEAN *distance function between two points or set of points, respectively.*

Closest Points Between Faces

Let f_1 and f_2 be disjoint faces of quadratic complexes that are embedded on the quadratic surfaces Q_1 and Q_2, where

$$Q_1 := \{\mathbf{x} \mid \mathbf{x}^T \mathbf{A} \mathbf{x} + 2\mathbf{a}^T \mathbf{x} + a_0 = 0\},$$
$$Q_2 := \{\mathbf{y} \mid \mathbf{y}^T \mathbf{B} \mathbf{y} + 2\mathbf{b}^T \mathbf{y} + b_0 = 0\}.$$

If $(\mathbf{p}_1, \mathbf{p}_2)$ is a pair of closest points between f_1 and f_2, then either

Closest Points Between Faces

Let f_1 and f_2 be disjoint faces of quadratic complexes that are embedded on the quadratic surfaces Q_1 and Q_2, where

$$\begin{aligned} Q_1 &:= \{\mathbf{x} \mid \mathbf{x}^T \mathbf{A} \mathbf{x} + 2\mathbf{a}^T \mathbf{x} + a_0 = 0\}, \\ Q_2 &:= \{\mathbf{y} \mid \mathbf{y}^T \mathbf{B} \mathbf{y} + 2\mathbf{b}^T \mathbf{y} + b_0 = 0\}. \end{aligned}$$

If $(\mathbf{p}_1, \mathbf{p}_2)$ is a pair of closest points between f_1 and f_2, then either

(i) $(\mathbf{p}_1, \mathbf{p}_2)$ is an extremum of the distance function between Q_1 and Q_2, i.e. there are $\alpha, \beta \in \mathbb{R},\ \alpha, \beta \neq 0$ s.t.

$$\mathbf{n}(\mathbf{p}_1) = \alpha(\mathbf{p}_2 - \mathbf{p}_1) \qquad \mathbf{n}(\mathbf{p}_2) = \beta(\mathbf{p}_1 - \mathbf{p}_2),$$

where $\mathbf{n}(\mathbf{p}_i)$ denotes the normal of Q_i in $\mathbf{p}_i$, or

Closest Points Between Faces

Let f_1 and f_2 be disjoint faces of quadratic complexes that are embedded on the quadratic surfaces Q_1 and Q_2, where

$$Q_1 := \{\mathbf{x} \mid \mathbf{x}^T \mathbf{A} \mathbf{x} + 2\mathbf{a}^T \mathbf{x} + a_0 = 0\},$$
$$Q_2 := \{\mathbf{y} \mid \mathbf{y}^T \mathbf{B} \mathbf{y} + 2\mathbf{b}^T \mathbf{y} + b_0 = 0\}.$$

If $(\mathbf{p}_1, \mathbf{p}_2)$ is a pair of closest points between f_1 and f_2, then either

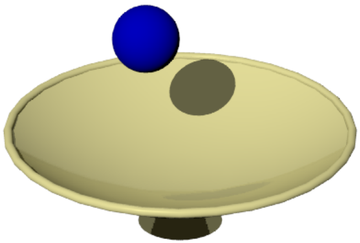
(i) $(\mathbf{p}_1, \mathbf{p}_2)$ is an extremum of the distance function between Q_1 and Q_2, i.e. there are $\alpha, \beta \in \mathbb{R},\ \alpha, \beta \neq 0$ s.t.

$$\mathbf{n}(\mathbf{p}_1) = \alpha(\mathbf{p}_2 - \mathbf{p}_1) \qquad \mathbf{n}(\mathbf{p}_2) = \beta(\mathbf{p}_1 - \mathbf{p}_2),$$

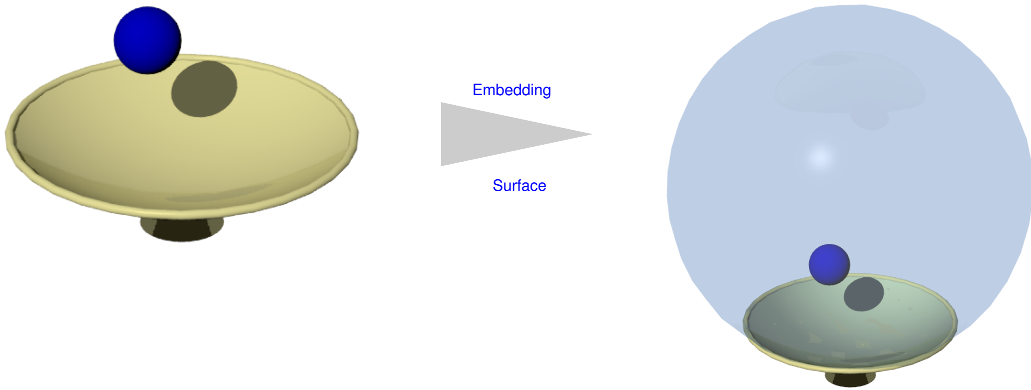
where $\mathbf{n}(\mathbf{p}_i)$ denotes the normal of Q_i in $\mathbf{p}_i$, or

(ii) $\mathbf{p}_1$ or $\mathbf{p}_2$ lies on the boundary of the face f_1 or f_2, respectively.

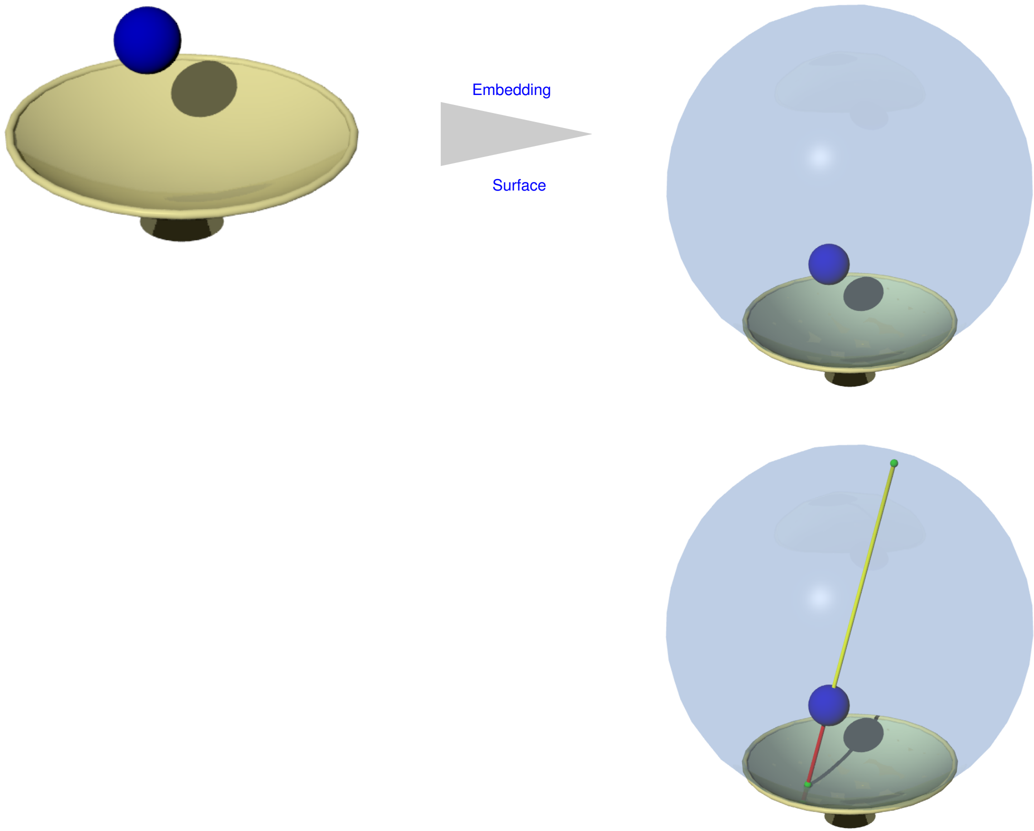
Distance Between Quadric Patches (Case I)



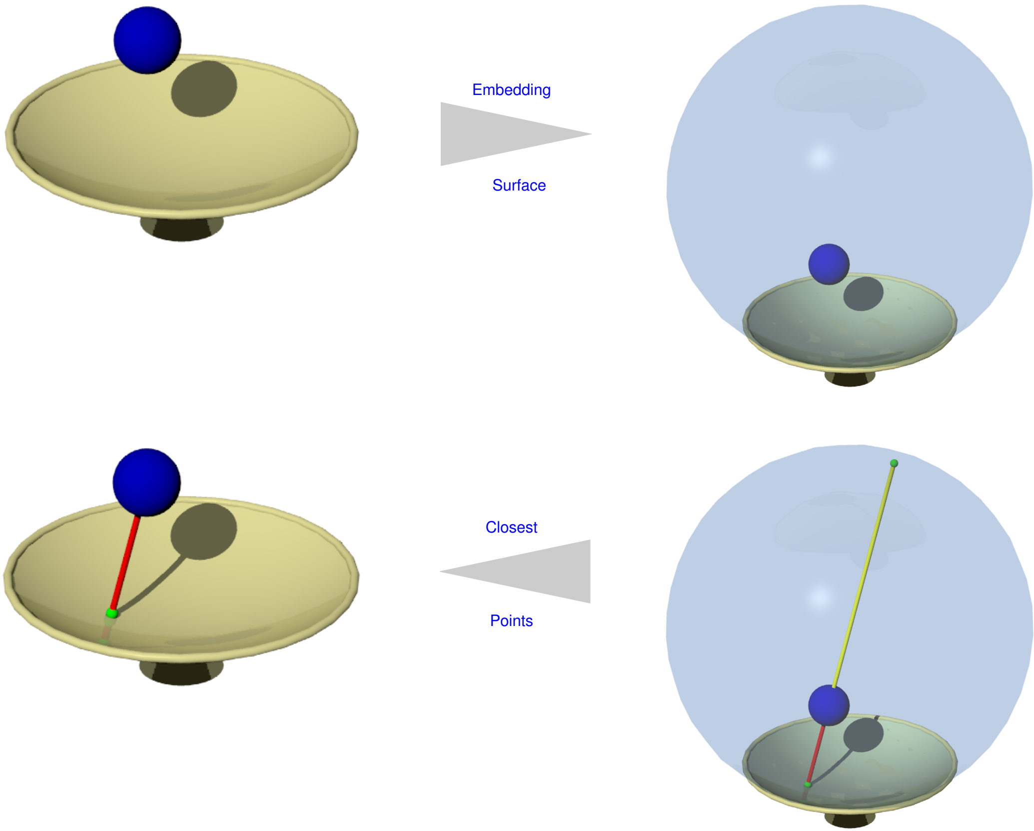
Distance Between Quadric Patches (Case I)



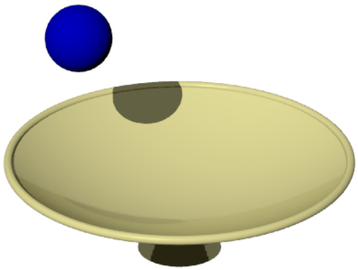
Distance Between Quadric Patches (Case I)



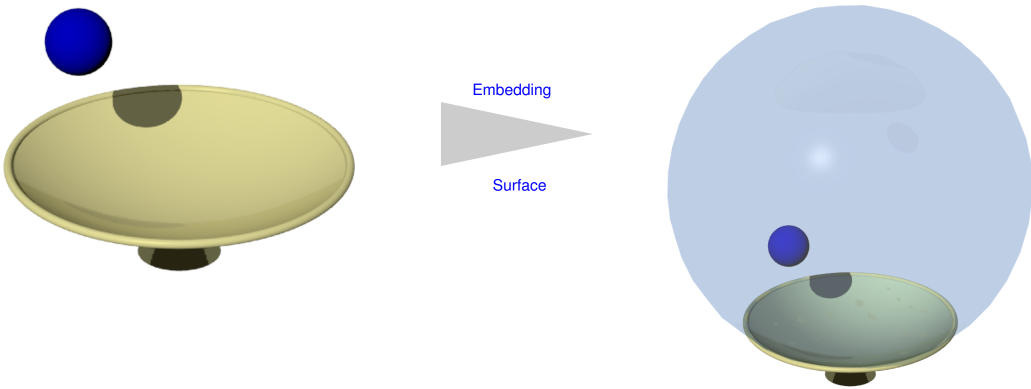
Distance Between Quadric Patches (Case I)



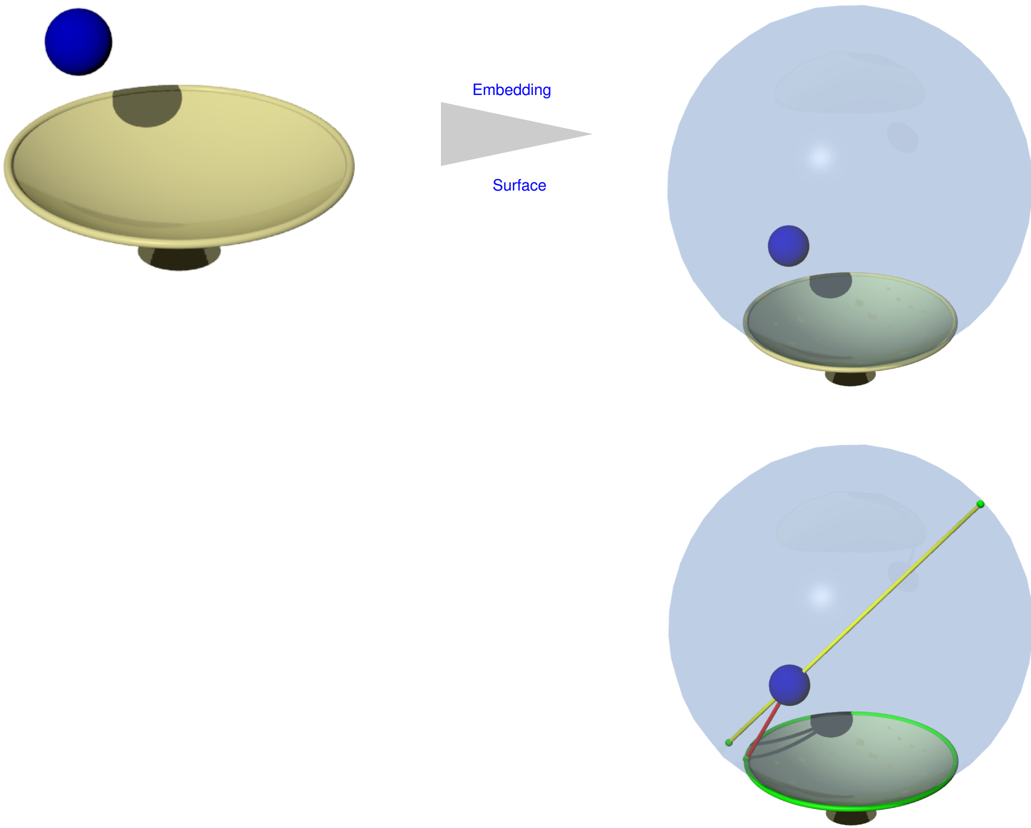
Distance Between Quadric Patches (Case II)



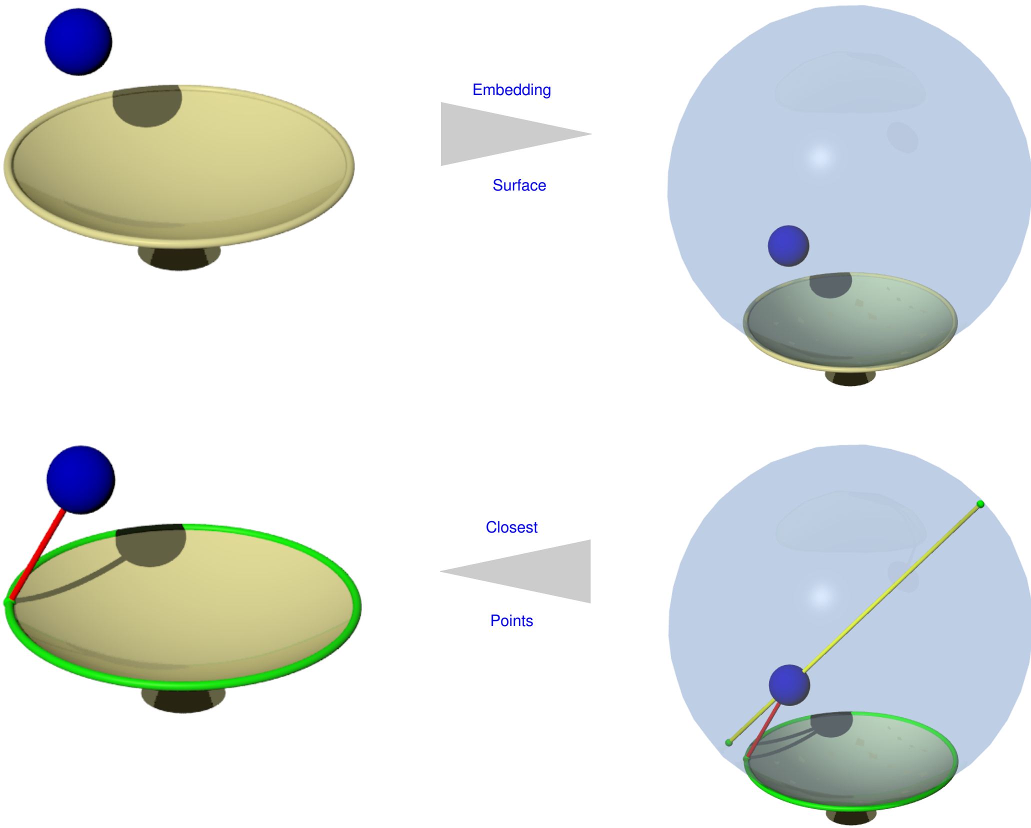
Distance Between Quadric Patches (Case II)



Distance Between Quadric Patches (Case II)



Distance Between Quadric Patches (Case II)



A Generic Algorithm

ENTITYDISTANCE(E_1, E_2)
(1)

A Generic Algorithm

$\textsc{EntityDistance}(E_1, E_2)$
(1) $\left[\mathrm{isDisjoint}, (\mathbf{p}_1, \mathbf{p}_2)\right] \leftarrow \textsc{Intersect}(E_1, E_2)$
(2) **if** $\mathrm{isDisjoint} = \mathrm{false}$
(3) **return** $\left[0, (\mathbf{p}_1, \mathbf{p}_2)\right]$

A Generic Algorithm

ENTITYDISTANCE(E_1, E_2)

(1) $\left[\mathrm{isDisjoint}, (\mathbf{p}_1, \mathbf{p}_2)\right] \leftarrow$ INTERSECT(E_1, E_2)

(2) **if** $\mathrm{isDisjoint} = \mathrm{false}$

(3) **return** $\left[0, (\mathbf{p}_1, \mathbf{p}_2)\right]$

(4) $\delta_G \leftarrow \infty$

(5) **while** $\left[\delta, (\mathbf{q}_1, \mathbf{q}_2)\right] \leftarrow$ EXTREMA(E_1, E_2)

(6) **if** $(\mathbf{q}_1 \in E_1)$ **and** $(\mathbf{q}_2 \in E_2)$

(7) **if** $\delta < \delta_G$

(8) $\delta_G \leftarrow \delta, \quad (\mathbf{p}_1, \mathbf{p}_2) \leftarrow (\mathbf{q}_1, \mathbf{q}_2)$

A Generic Algorithm

$\text{EntityDistance}(E_1, E_2)$

$\left[\mathrm{isDisjoint}, (\mathbf{p}_1, \mathbf{p}_2)\right] \leftarrow \text{Intersect}(E_1, E_2)$

if $\mathrm{isDisjoint} = \mathrm{false}$

  return $\left[0, (\mathbf{p}_1, \mathbf{p}_2)\right]$

$\delta_G \leftarrow \infty$

while $\left[\delta, (\mathbf{q}_1, \mathbf{q}_2)\right] \leftarrow \text{Extrema}(E_1, E_2)$

  if $(\mathbf{q}_1 \in E_1)$ **and** $(\mathbf{q}_2 \in E_2)$

    if $\delta < \delta_G$

      $\delta_G \leftarrow \delta, \quad (\mathbf{p}_1, \mathbf{p}_2) \leftarrow (\mathbf{q}_1, \mathbf{q}_2)$

foreach subentity E **of** E_1

  $\left[\delta, (\mathbf{q}_1, \mathbf{q}_2)\right] \leftarrow \text{EntityDistance}(E, E_2)$

  if $\delta < \delta_G$

    $\delta_G \leftarrow \delta, \quad (\mathbf{p}_1, \mathbf{p}_2) \leftarrow (\mathbf{q}_1, \mathbf{q}_2)$

A Generic Algorithm

$\textsc{EntityDistance}(E_1, E_2)$

$\quad [\mathrm{isDisjoint}, (\mathbf{p}_1, \mathbf{p}_2)] \leftarrow \textsc{Intersect}(E_1, E_2)$

$\quad$ **if** $\mathrm{isDisjoint} = \mathrm{false}$

$\quad\quad$ **return** $[0, (\mathbf{p}_1, \mathbf{p}_2)]$

$\quad \delta_G \leftarrow \infty$

$\quad$ **while** $[\delta, (\mathbf{q}_1, \mathbf{q}_2)] \leftarrow \textsc{Extrema}(E_1, E_2)$

$\quad\quad$ **if** $(\mathbf{q}_1 \in E_1)$ **and** $(\mathbf{q}_2 \in E_2)$

$\quad\quad\quad$ **if** $\delta < \delta_G$

$\quad\quad\quad\quad \delta_G \leftarrow \delta, \quad (\mathbf{p}_1, \mathbf{p}_2) \leftarrow (\mathbf{q}_1, \mathbf{q}_2)$

$\quad$ **foreach subentity** E **of** E_1

$\quad\quad [\delta, (\mathbf{q}_1, \mathbf{q}_2)] \leftarrow \textsc{EntityDistance}(E, E_2)$

$\quad\quad$ **if** $\delta < \delta_G$

$\quad\quad\quad \delta_G \leftarrow \delta, \quad (\mathbf{p}_1, \mathbf{p}_2) \leftarrow (\mathbf{q}_1, \mathbf{q}_2)$

$\quad$ **foreach subentity** E **of** E_2

$\quad\quad [\delta, (\mathbf{q}_1, \mathbf{q}_2)] \leftarrow \textsc{EntityDistance}(E_1, E)$

$\quad\quad$ **if** $\delta < \delta_G$

$\quad\quad\quad \delta_G \leftarrow \delta, \quad (\mathbf{p}_1, \mathbf{p}_2) \leftarrow (\mathbf{q}_1, \mathbf{q}_2)$

A Generic Algorithm

EntityDistance(E_1, E_2)

$[\mathrm{isDisjoint}, (\mathbf{p}_1, \mathbf{p}_2)] \leftarrow$ Intersect(E_1, E_2)

if $\mathrm{isDisjoint} = \mathrm{false}$

  return $[0, (\mathbf{p}_1, \mathbf{p}_2)]$

$\delta_G \leftarrow \infty$

while $[\delta, (\mathbf{q}_1, \mathbf{q}_2)] \leftarrow$ Extrema(E_1, E_2)

  if $(\mathbf{q}_1 \in E_1)$ **and** $(\mathbf{q}_2 \in E_2)$

    if $\delta < \delta_G$

      $\delta_G \leftarrow \delta, \quad (\mathbf{p}_1, \mathbf{p}_2) \leftarrow (\mathbf{q}_1, \mathbf{q}_2)$

foreach subentity E **of** E_1

  $[\delta, (\mathbf{q}_1, \mathbf{q}_2)] \leftarrow$ EntityDistance(E, E_2)

  if $\delta < \delta_G$

    $\delta_G \leftarrow \delta, \quad (\mathbf{p}_1, \mathbf{p}_2) \leftarrow (\mathbf{q}_1, \mathbf{q}_2)$

foreach subentity E **of** E_2

  $[\delta, (\mathbf{q}_1, \mathbf{q}_2)] \leftarrow$ EntityDistance(E_1, E)

  if $\delta < \delta_G$

    $\delta_G \leftarrow \delta, \quad (\mathbf{p}_1, \mathbf{p}_2) \leftarrow (\mathbf{q}_1, \mathbf{q}_2)$

return $[\delta_G, (\mathbf{p}_1, \mathbf{p}_2)]$

Degree Complexity of the Polynomial Systems

Theorem 1 (General Quadratic Complexes)

- *The distance between two faces of quadratic complexes can be computed by solving systems of univariate and bivariate polynomials in which the degree of every variable is at most 6.*
- *These systems can be solved by finding the roots of univariate polynomials of a degree of at most 24.*

Degree Complexity of the Polynomial Systems

Theorem 1 (General Quadratic Complexes)

- *The distance between two faces of quadratic complexes can be computed by solving systems of univariate and bivariate polynomials in which the degree of every variable is at most 6.*
- *These systems can be solved by finding the roots of univariate polynomials of a degree of at most 24.*

Theorem 2 (Natural Quadratic Complexes)
The distance between two faces embedded on natural quadrics and trimmed by natural conics can be computed by solving univariate polynomials of a degree of at most 8.

Degree Complexity of the Polynomial Systems

Theorem 1 (General Quadratic Complexes)

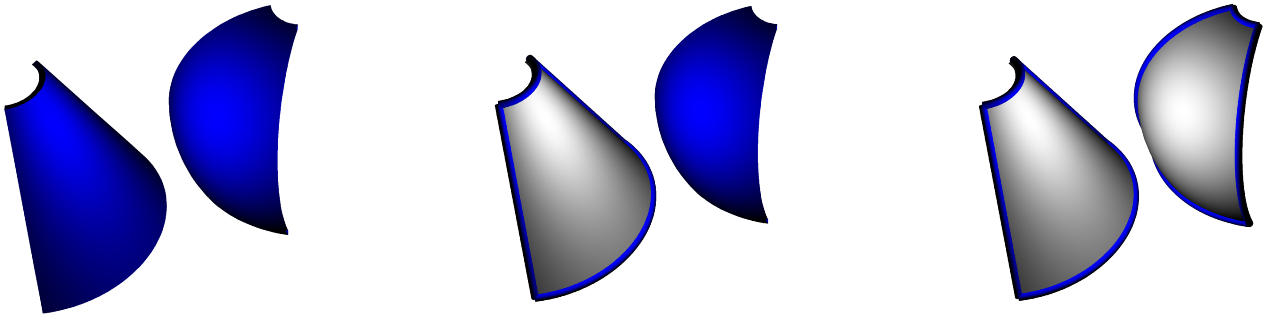
- *The distance between two faces of quadratic complexes can be computed by solving systems of univariate and bivariate polynomials in which the degree of every variable is at most 6.*
- *These systems can be solved by finding the roots of univariate polynomials of a degree of at most 24.*

Theorem 2 (Natural Quadratic Complexes)
The distance between two faces embedded on natural quadrics and trimmed by natural conics can be computed by solving univariate polynomials of a degree of at most 8.

Remark 1 (Torus)
If one extends the classes by the torus, the results remain valid. The distance to any other surface or curve can be computed by considering its main circle.

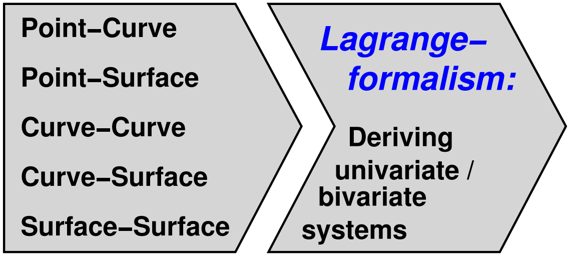
Overview of the Approach

- Point–Curve
- Point–Surface
- Curve–Curve
- Curve–Surface
- Surface–Surface

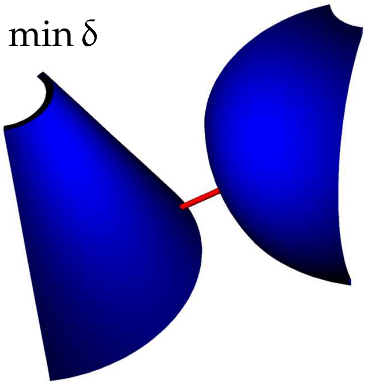


Surface-Surface   Edge-Surface   Edge-Edge   ...

Overview of the Approach



min δ



LAGRANGE

formalism

$$f(x,y) = \sum_{i,j} a_{ij} x^i y^j = 0$$

$$g(x,y) = \sum_{i,j} b_{ij} x^i y^j = 0$$

Minimization problem

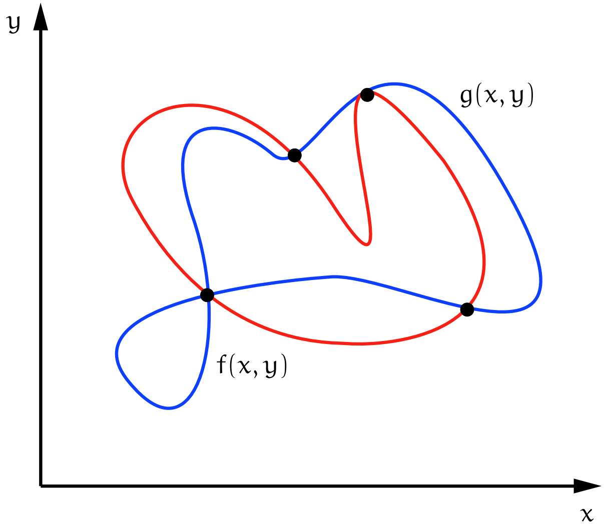
System of equations

Overview of the Approach

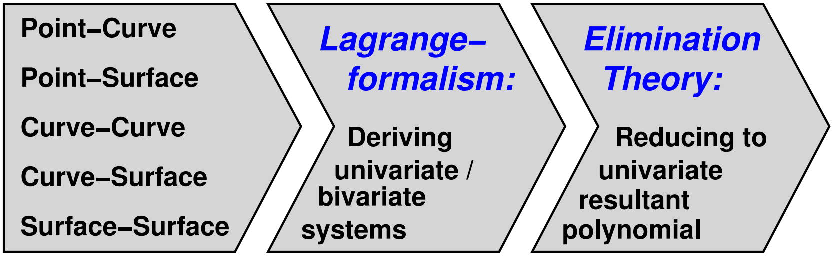
Point–Curve
Point–Surface
Curve–Curve
Curve–Surface
Surface–Surface

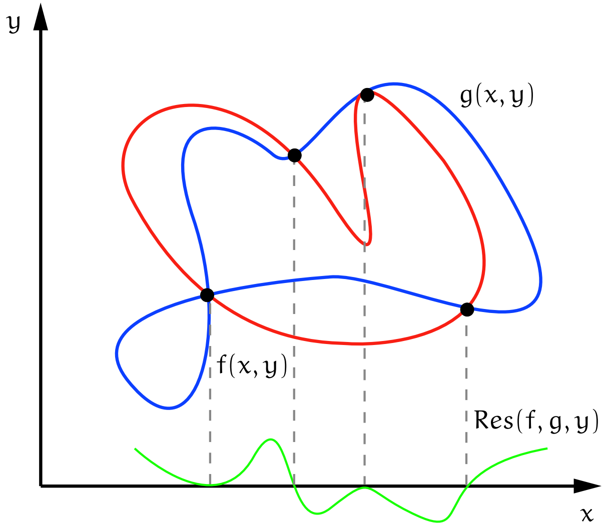
Lagrange–formalism:
Deriving univariate / bivariate systems

Elimination Theory:
Reducing to univariate resultant polynomial

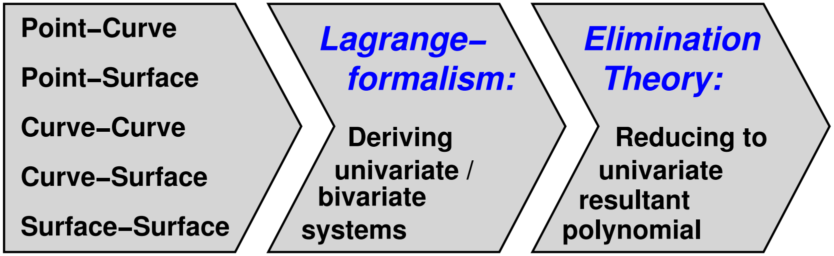


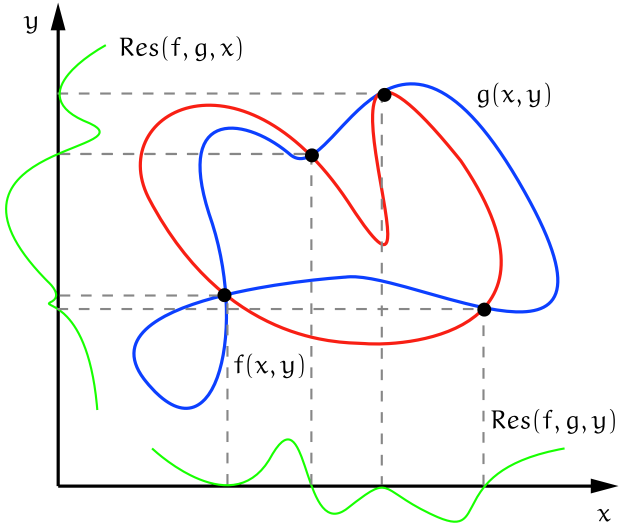
Overview of the Approach



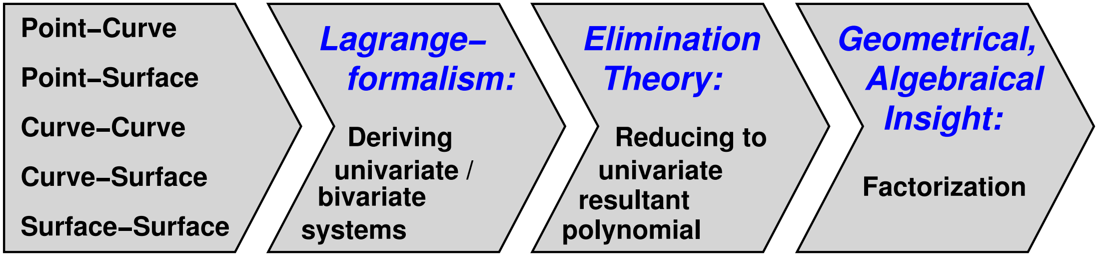


Overview of the Approach





Overview of the Approach

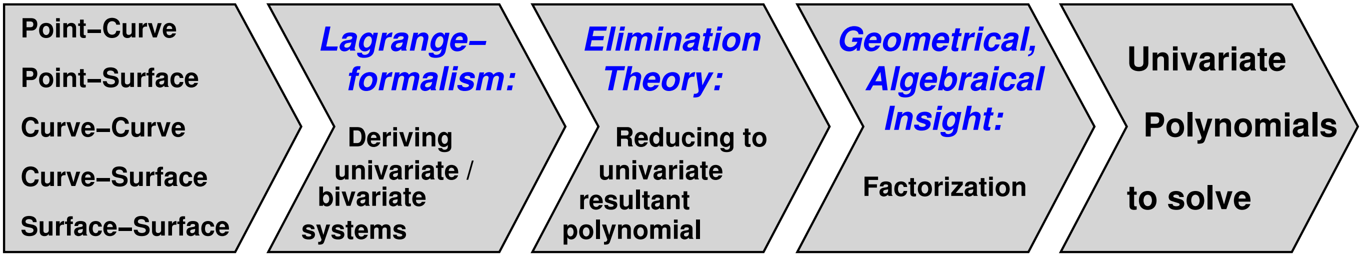


Factorization:

$$\mathrm{Res}(f, g, x) \;=\; p_x(y) \cdot q_x(y)$$

$$\mathrm{Res}(f, g, y) \;=\; p_y(x) \cdot q_y(x)$$

Overview of the Approach



Univariate Polynomials to solve:

$$p_x(y) = 0 \qquad q_x(y) = 0$$

$$p_y(x) = 0 \qquad q_y(x) = 0$$

The Surface-Surface Case

By setting up the LAGRANGE formalism for the problem

$$\min\,(\boldsymbol{x}-\boldsymbol{y})^2, \quad \boldsymbol{x} \in Q_1, \boldsymbol{y} \in Q_2$$

The Surface-Surface Case

By setting up the LAGRANGE formalism for the problem

$$\min\,(\mathbf{x}-\mathbf{y})^2, \quad \mathbf{x} \in Q_1, \mathbf{y} \in Q_2$$

we get the LAGRANGE function $\mathcal{L}$ and conditions $(i), \ldots, (iv)$:

$$\begin{aligned}\mathcal{L}(\mathbf{x}, \mathbf{y}; \alpha, \beta) &= (\mathbf{x}-\mathbf{y})^2 + \alpha(\mathbf{x}^T\mathbf{A}\mathbf{x} + 2\mathbf{a}^T\mathbf{x} + a_0) \\ &\quad + \beta(\mathbf{y}^T\mathbf{B}\mathbf{y} + 2\mathbf{b}^T\mathbf{y} + b_0)\end{aligned}$$

The Surface-Surface Case

By setting up the LAGRANGE formalism for the problem

$$\min\,(\mathbf{x}-\mathbf{y})^2, \quad \mathbf{x} \in Q_1, \mathbf{y} \in Q_2$$

we get the LAGRANGE function $\mathcal{L}$ and conditions $(i), \ldots, (iv)$:

$$\begin{aligned}
\mathcal{L}(\mathbf{x},\mathbf{y};\alpha,\beta) &= (\mathbf{x}-\mathbf{y})^2 + \alpha(\mathbf{x}^T\mathbf{A}\mathbf{x} + 2\mathbf{a}^T\mathbf{x} + a_0) \\
&\quad + \beta(\mathbf{y}^T\mathbf{B}\mathbf{y} + 2\mathbf{b}^T\mathbf{y} + b_0) \\
(i)\ \partial\frac{\mathcal{L}(.)}{\partial\mathbf{x}} &= 0 \quad \Longleftrightarrow \quad \alpha(\mathbf{A}\mathbf{x}+\mathbf{a}) = \mathbf{y}-\mathbf{x} \\
(ii)\ \partial\frac{\mathcal{L}(.)}{\partial\mathbf{y}} &= 0 \quad \Longleftrightarrow \quad \beta(\mathbf{B}\mathbf{y}+\mathbf{b}) = \mathbf{x}-\mathbf{y}
\end{aligned}$$

The Surface-Surface Case

By setting up the LAGRANGE formalism for the problem

$$\min\,(\mathbf{x}-\mathbf{y})^2, \quad \mathbf{x} \in Q_1, \mathbf{y} \in Q_2$$

we get the LAGRANGE function $\mathcal{L}$ and conditions $(i), \ldots, (iv)$:

$$\begin{aligned}
\mathcal{L}(\mathbf{x},\mathbf{y};\alpha,\beta) &= (\mathbf{x}-\mathbf{y})^2 + \alpha(\mathbf{x}^T\mathbf{A}\mathbf{x} + 2\mathbf{a}^T\mathbf{x} + a_0) \\
&\quad + \beta(\mathbf{y}^T\mathbf{B}\mathbf{y} + 2\mathbf{b}^T\mathbf{y} + b_0) \\
(i)\ \partial\frac{\mathcal{L}(.)}{\partial\mathbf{x}} &= 0 \quad \Longleftrightarrow \quad \alpha(\mathbf{A}\mathbf{x}+\mathbf{a}) = \mathbf{y}-\mathbf{x} \\
(ii)\ \partial\frac{\mathcal{L}(.)}{\partial\mathbf{y}} &= 0 \quad \Longleftrightarrow \quad \beta(\mathbf{B}\mathbf{y}+\mathbf{b}) = \mathbf{x}-\mathbf{y} \\
(iii)\ \partial\frac{\mathcal{L}(.)}{\partial\alpha} &= 0 \quad \Longleftrightarrow \quad \mathbf{x}^T A\mathbf{x} + 2\mathbf{a}^T\mathbf{x} + a_0 = 0 \\
(iv)\ \partial\frac{\mathcal{L}(.)}{\partial\beta} &= 0 \quad \Longleftrightarrow \quad \mathbf{y}^T B\mathbf{y} + 2\mathbf{b}^T\mathbf{y} + b_0 = 0
\end{aligned}$$

Solving the Lagrange System

By setting $\lambda := 1/\alpha$ and $\mu := 1/\beta$ we can derive from (i) and (ii):

$$\begin{aligned} \mathbf{x} &= -(\mathbf{B}\mathbf{A} + \lambda\mathbf{B} + \mu\mathbf{A})^{-1}(\mathbf{B}\mathbf{a} + \lambda\mathbf{b} + \mu\mathbf{a}) =: -\frac{\overline{\mathbf{C}}_{\lambda,\mu}}{|\mathbf{C}_{\lambda,\mu}|}\mathbf{c}_B, \\ \mathbf{y} &= -(\mathbf{A}\mathbf{B} + \lambda\mathbf{B} + \mu\mathbf{A})^{-1}(\mathbf{A}\mathbf{b} + \lambda\mathbf{b} + \mu\mathbf{a}) =: -\frac{\overline{\mathbf{C}}_{\lambda,\mu}^{T}}{|\mathbf{C}_{\lambda,\mu}|}\mathbf{c}_A, \end{aligned}$$

where $\overline{\mathbf{C}}_{\lambda,\mu}$ and $|\mathbf{C}_{\lambda,\mu}|$ are (matrix) polynomials in λ and μ.

Solving the Lagrange System

By setting $\lambda := 1/\alpha$ and $\mu := 1/\beta$ we can derive from (i) and (ii):

$$\mathbf{x} = -(\mathbf{BA} + \lambda\mathbf{B} + \mu\mathbf{A})^{-1}(\mathbf{Ba} + \lambda\mathbf{b} + \mu\mathbf{a}) =: -\frac{\overline{\mathbf{C}}_{\lambda,\mu}}{|\mathbf{C}_{\lambda,\mu}|}\mathbf{c}_B,$$

$$\mathbf{y} = -(\mathbf{AB} + \lambda\mathbf{B} + \mu\mathbf{A})^{-1}(\mathbf{Ab} + \lambda\mathbf{b} + \mu\mathbf{a}) =: -\frac{\overline{\mathbf{C}}^T_{\lambda,\mu}}{|\mathbf{C}_{\lambda,\mu}|}\mathbf{c}_A,$$

where $\overline{\mathbf{C}}_{\lambda,\mu}$ and $|\mathbf{C}_{\lambda,\mu}|$ are (matrix) polynomials in λ and μ.

Substituting $\mathbf{x}$ and $\mathbf{y}$ in (iii) and (iv) and multiplying my the denominator, gives the system:

$$f(\lambda,\mu) = \mathbf{c}_B^T\overline{\mathbf{C}}^T_{\lambda,\mu}\mathbf{A}\overline{\mathbf{C}}_{\lambda,\mu}\mathbf{c}_B - 2|\mathbf{C}_{\lambda,\mu}|\mathbf{a}^T\overline{\mathbf{C}}_{\lambda,\mu}\mathbf{c}_B + a_0|\mathbf{C}_{\lambda,\mu}|^2 = 0,$$

$$g(\lambda,\mu) = \mathbf{c}_A^T\overline{\mathbf{C}}_{\lambda,\mu}\mathbf{B}\overline{\mathbf{C}}^T_{\lambda,\mu}\mathbf{c}_A - 2|\mathbf{C}_{\lambda,\mu}|\mathbf{b}^T\overline{\mathbf{C}}^T_{\lambda,\mu}\mathbf{c}_A + b_0|\mathbf{C}_{\lambda,\mu}|^2 = 0,$$

The Inverse of $\mathbf{C}_{\lambda,\mu}$

Lemma 1
The adjoint and determinant of $\mathbf{C}_{\lambda,\mu} = \mathbf{B}\mathbf{A} + \lambda\mathbf{B} + \mu\mathbf{A}$ *is given by*

$$\begin{aligned}
\overline{\mathbf{C}_{\lambda,\mu}} &= \overline{\mathbf{B}}\lambda^2 + \overline{\mathbf{A}}\mu^2 + \mathbf{T}_A\overline{\mathbf{B}}\lambda + \overline{\mathbf{A}}\mathbf{T}_B\mu + (\mathbf{T}_B\mathbf{T}_A - \mathbf{T}_{AB})\lambda\mu + \overline{\mathbf{A}}\,\overline{\mathbf{B}}, \\
|\mathbf{C}_{\lambda,\mu}| &= |\mathbf{B}|\lambda^3 + |\mathbf{A}|\mu^3 + |\mathbf{B}|tr(\mathbf{A})\lambda^2 + |\mathbf{A}|tr(\mathbf{B})\mu^2 + \\
&\quad |\mathbf{B}|tr(\overline{\mathbf{A}})\lambda + |\mathbf{A}|tr(\overline{\mathbf{B}})\mu + tr(\overline{\mathbf{B}}\mathbf{A})\lambda^2\mu + tr(\overline{\mathbf{A}}\mathbf{B})\lambda\mu^2 + \\
&\quad \big(tr(\overline{\mathbf{A}})tr(\overline{\mathbf{B}}) - tr(\overline{\mathbf{A}}\,\overline{\mathbf{B}})\big)\lambda\mu + |\mathbf{A}||\mathbf{B}|,
\end{aligned}$$

where $\mathbf{T}_M := tr(\mathbf{M})\mathbf{E} - \mathbf{M}$ *for a matrix* $\mathbf{M} \in \mathbb{R}^{3\times 3}$.

The Inverse of $\mathbf{C}_{\lambda,\mu}$

Lemma 1
The adjoint and determinant of $\mathbf{C}_{\lambda,\mu} = \mathbf{B}\mathbf{A} + \lambda\mathbf{B} + \mu\mathbf{A}$ *is given by*

$$\begin{aligned}
\overline{\mathbf{C}_{\lambda,\mu}} &= \overline{\mathbf{B}}\lambda^2 + \overline{\mathbf{A}}\mu^2 + \mathbf{T}_A\overline{\mathbf{B}}\lambda + \overline{\mathbf{A}}\mathbf{T}_B\mu + (\mathbf{T}_B\mathbf{T}_A - \mathbf{T}_{AB})\lambda\mu + \overline{\mathbf{A}}\,\overline{\mathbf{B}}, \\
|\mathbf{C}_{\lambda,\mu}| &= |\mathbf{B}|\lambda^3 + |\mathbf{A}|\mu^3 + |\mathbf{B}|tr(\mathbf{A})\lambda^2 + |\mathbf{A}|tr(\mathbf{B})\mu^2 + \\
&\quad |\mathbf{B}|tr(\overline{\mathbf{A}})\lambda + |\mathbf{A}|tr(\overline{\mathbf{B}})\mu + tr(\overline{\mathbf{B}}\mathbf{A})\lambda^2\mu + tr(\overline{\mathbf{A}}\mathbf{B})\lambda\mu^2 + \\
&\quad \big(tr(\overline{\mathbf{A}})tr(\overline{\mathbf{B}}) - tr(\overline{\mathbf{A}}\,\overline{\mathbf{B}})\big)\lambda\mu + |\mathbf{A}||\mathbf{B}|,
\end{aligned}$$

where $\mathbf{T}_M := tr(\mathbf{M})\mathbf{E} - \mathbf{M}$ *for a matrix* $\mathbf{M} \in \mathbb{R}^{3\times 3}$.

Proposition 1 (Bivariate Degree Complexity)
The polynomials f *and* g *have degree 6 in* λ *as well as* μ.
Moreover the total degree of f *and* g *is also 6.*

The Inverse of $C_{\lambda,\mu}$

Lemma 1
The adjoint and determinant of $\mathbf{C}_{\lambda,\mu} = \mathbf{B}\mathbf{A} + \lambda\mathbf{B} + \mu\mathbf{A}$ *is given by*

$$\begin{aligned}
\overline{\mathbf{C}_{\lambda,\mu}} &= \overline{\mathbf{B}}\lambda^2 + \overline{\mathbf{A}}\mu^2 + \mathsf{T}_A\overline{\mathbf{B}}\lambda + \overline{\mathbf{A}}\mathsf{T}_B\mu + (\mathsf{T}_B\mathsf{T}_A - \mathsf{T}_{AB})\lambda\mu + \overline{\mathbf{A}}\,\overline{\mathbf{B}}, \\
|\mathbf{C}_{\lambda,\mu}| &= |\mathbf{B}|\lambda^3 + |\mathbf{A}|\mu^3 + |\mathbf{B}|tr(\mathbf{A})\lambda^2 + |\mathbf{A}|tr(\mathbf{B})\mu^2 + \\
&\quad |\mathbf{B}|tr(\overline{\mathbf{A}})\lambda + |\mathbf{A}|tr(\overline{\mathbf{B}})\mu + tr(\overline{\mathbf{B}}\mathbf{A})\lambda^2\mu + tr(\overline{\mathbf{A}}\mathbf{B})\lambda\mu^2 + \\
&\quad \left(tr(\overline{\mathbf{A}})tr(\overline{\mathbf{B}}) - tr(\overline{\mathbf{A}}\,\overline{\mathbf{B}})\right)\lambda\mu + |\mathbf{A}||\mathbf{B}|,
\end{aligned}$$

where $\mathsf{T}_M := tr(\mathbf{M})\mathsf{E} - \mathbf{M}$ *for a matrix* $\mathbf{M} \in \mathbb{R}^{3\times 3}$.

Proposition 1 (Bivariate Degree Complexity)
The polynomials f *and* g *have degree 6 in* λ *as well as* μ.
Moreover the total degree of f *and* g *is also 6.*

Corollary 1 (Bezout)
The degree of the resultant polynomial $\mathrm{Res}(f, g)$ *is at most 36.*

Factorization of the Resultant Polynomial

Lemma 2
Let $f = g = 0$ *be our system of polynomial equations, i.e.*

$$
\begin{aligned}
f(\lambda,\mu) &= \mathbf{c}_B^T \overline{\mathbf{C}}_{\lambda,\mu}^T \mathbf{A} \overline{\mathbf{C}}_{\lambda,\mu} \mathbf{c}_B - 2|\mathbf{C}_{\lambda,\mu}| \mathbf{a}^T \overline{\mathbf{C}}_{\lambda,\mu} \mathbf{c}_B + a_0 |\mathbf{C}_{\lambda,\mu}|^2 = 0, \\
g(\lambda,\mu) &= \mathbf{c}_A^T \overline{\mathbf{C}}_{\lambda,\mu} \mathbf{B} \overline{\mathbf{C}}_{\lambda,\mu}^T \mathbf{c}_A - 2|\mathbf{C}_{\lambda,\mu}| \mathbf{b}^T \overline{\mathbf{C}}_{\lambda,\mu}^T \mathbf{c}_A + b_0 |\mathbf{C}_{\lambda,\mu}|^2 = 0,
\end{aligned}
$$

and the system $\mathbf{h}$ *be defined as follows:*

$$
\mathbf{h}(\lambda,\mu) := (h_1, h_2, h_3)^T = \overline{\mathbf{C}}_{\lambda,\mu} \mathbf{c}_B = \mathbf{0}.
$$

Then the common roots of the polynomials $r_{ij} := \mathrm{Res}(h_i, h_j)$*,* $1 \le i < j \le 3$*, solve* $\mathrm{Res}(f, g)$ *with multiplicity 4.*

Factorization of the Resultant Polynomial

Lemma 2
Let $f = g = 0$ *be our system of polynomial equations, i.e.*

$$
\begin{aligned}
f(\lambda,\mu) &= \mathbf{c}_B^T \overline{\mathbf{C}}_{\lambda,\mu}^T \mathbf{A} \overline{\mathbf{C}}_{\lambda,\mu} \mathbf{c}_B - 2|\mathbf{C}_{\lambda,\mu}| \mathbf{a}^T \overline{\mathbf{C}}_{\lambda,\mu} \mathbf{c}_B + a_0 |\mathbf{C}_{\lambda,\mu}|^2 = 0, \\
g(\lambda,\mu) &= \mathbf{c}_A^T \overline{\mathbf{C}}_{\lambda,\mu} \mathbf{B} \overline{\mathbf{C}}_{\lambda,\mu}^T \mathbf{c}_A - 2|\mathbf{C}_{\lambda,\mu}| \mathbf{b}^T \overline{\mathbf{C}}_{\lambda,\mu}^T \mathbf{c}_A + b_0 |\mathbf{C}_{\lambda,\mu}|^2 = 0,
\end{aligned}
$$

and the system $\mathbf{h}$ *be defined as follows:*

$$
\mathbf{h}(\lambda,\mu) := (h_1, h_2, h_3)^T = \overline{\mathbf{C}}_{\lambda,\mu} \mathbf{c}_B = \mathbf{0}.
$$

Then the common roots of the polynomials $r_{ij} := \mathrm{Res}(h_i, h_j)$, $1 \le i < j \le 3$, *solve* $\mathrm{Res}(f, g)$ *with multiplicity 4.*

Proposition 2 (Degree Complexity)
Let p *denote the polynomial given by the common roots of* r_{ij}, $1 \le i < j \le 3$, *and their multiplicities in* $\mathrm{Res}(f, g)$. *Then the remaining polynomial* $\mathrm{Res}(f, g)/p$ *is of a degree of at most* 24.

General Conics and Quadrics

	Point	Curve	Non-Central Surface	Central Surface
Point	1	4	5	6
Curve		16	16	20
Non-Central Surface			13	18
Central Surface				24

Natural Conics, Quadrics and the Torus

	Point	Curve	Non-Central Surface	Central Surface	Torus
Point	1	2	2	2	2
Curve		8	4	8	8
Non-Central Surface			2	2	4
Central Surface				4	8
Torus					8

Conclusions

Theoretical point of view:

Conclusions

Theoretical point of view:

- We have presented an exhaustive analysis of the distance computation problem between conics, quadrics and the torus.

Conclusions

Theoretical point of view:

- We have presented an exhaustive analysis of the distance computation problem between conics, quadrics and the torus.
- We use a uniform formalism for studying all special cases.

Conclusions

Theoretical point of view:

- We have presented an exhaustive analysis of the distance computation problem between conics, quadrics and the torus.
- We use a uniform formalism for studying all special cases.
- The degree results are provably optimal for the case of extended natural quadratic complexes.

Conclusions

Theoretical point of view:

- We have presented an exhaustive analysis of the distance computation problem between conics, quadrics and the torus.
- We use a uniform formalism for studying all special cases.
- The degree results are provably optimal for the case of extended natural quadratic complexes.

Practical point of view:

Conclusions

Theoretical point of view:

- We have presented an exhaustive analysis of the distance computation problem between conics, quadrics and the torus.
- We use a uniform formalism for studying all special cases.
- The degree results are provably optimal for the case of extended natural quadratic complexes.

Practical point of view:

- In contrast to standard numerical techniques the algebraic approach guarantees to find all extrema of the distance function.

Conclusions

Theoretical point of view:

- We have presented an exhaustive analysis of the distance computation problem between conics, quadrics and the torus.
- We use a uniform formalism for studying all special cases.
- The degree results are provably optimal for the case of extended natural quadratic complexes.

Practical point of view:

- In contrast to standard numerical techniques the algebraic approach guarantees to find all extrema of the distance function.
- Interval methods break down due to numerical difficulties and singularities of the occuring bivariate polynomial systems.

Conclusions

Theoretical point of view:

- We have presented an exhaustive analysis of the distance computation problem between conics, quadrics and the torus.
- We use a uniform formalism for studying all special cases.
- The degree results are provably optimal for the case of extended natural quadratic complexes.

Practical point of view:

- In contrast to standard numerical techniques the algebraic approach guarantees to find all extrema of the distance function.
- Interval methods break down due to numerical difficulties and singularities of the occuring bivariate polynomial systems.
- Using floating-point arithmetic the approach shows sufficient numerical stability as well as suitability for real-time applications.

Conclusions

Theoretical point of view:

- We have presented an exhaustive analysis of the distance computation problem between conics, quadrics and the torus.
- We use a uniform formalism for studying all special cases.
- The degree results are provably optimal for the case of extended natural quadratic complexes.

Practical point of view:

- In contrast to standard numerical techniques the algebraic approach guarantees to find all extrema of the distance function.
- Interval methods break down due to numerical difficulties and singularities of the occuring bivariate polynomial systems.
- Using floating-point arithmetic the approach shows sufficient numerical stability as well as suitability for real-time applications.
- Inaccurate solutions can be efficiently polished using NEWTON iterations.