Conic Polygons

Boolean Operations on Polygons with Conic Arcs

Eric Berberich
Arno Eigenwillig
Michael Hemmer
Susan Hert
Michael Kerber
Lutz Kettner
Kurt Mehlhorn
Michael Sagraloff
Elmar Schömer
Nicola Wolpert
Goals

• conic = zero set of a quadratic equation
  \[ Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0 \]

• conic point = intersection of two conics

• conic arc = part of a conic between two conic points

• conic polygon = polygon with conic arcs

• compute
  • arrangements of conic arcs
  • boolean operations on conic polygons

• want
  • mathematically correct result
  • for all inputs (arbitrary degeneracies)
  • with reasonable efficiency

• in this talk: how we reached the goal
Union of Two Polygons

polygons with circular arcs

green polygon is union of red and blue polygon

computation of union took one minute for two roses with 1000 petals in 2002
by 2008, we are below one second
Arrangement of Ellipses

Implementation can handle arbitrary degeneracies

- many curves have a common point
- different slopes
- same slope, different curvature,
- same slope and curvature, diff . . .

Computation took about 15 seconds in 2002; by 2008, below one second
Overview

- it suffices to compute arrangements of conic arcs
- once the arrangement is known, everything is as in the case of straight-line polygons (LEDA code worked without change)
- Bentley-Ottmann sweep works for arbitrary curves
  - except for one little detail
  - predicates require thought and make the difference
- optimizations
- implementations are part of EXACUS


- for more recent work see the home pages of Eric Berberich, Arno Eigenwillig, Michael Hemmer, Michael Kerber, Kurt Mehlhorn, and Michael Sagraloff
Bentley-Ottmann Sweep for Line Intersection

**input:** a set of line segments

**output:** the planar map (arrangement) $G$ defined by the segments; $G$ has one vertex for each endpoint and each intersection

- sweep a vertical line $L$ across the plane and maintain
- Y-structure = sorted sequence of intersections between $L$ and given curves
- X-structure = already known vertices ahead of sweep line
- update at event points
- $G$ emerges to the left of $L$
Bentley-Ottmann Sweep II

- events are processed in lexicographic order of \((x, y)\)
- at an event point:
  - create a new vertex \(v\) in \(G\)
  - remove all ending segments from the Y-structure (if any) and update \(G\)
  - compute the new order of the segments passing through the vertex (if any) and update \(G\)
  - if only starting segments, insert the start point into the Y-structure and update \(G\)
  - insert the starting segments into the Y-structure
  - compute intersection between segments that became adjacent
Predicates and Functions Required for the Sweep

- given two conics $C$ and $D$, compute their intersection points

- lex-order on intersection points and starting and endpoints for order of X-structure

- (above, on, below)-comparison of points and arcs for order of Y-structure

- multiplicity of intersections for sweeping through a vertex

- I will only discuss the first and the last item

- I will bore you with details because the details make all the difference

- I will discuss conics and at the end hint at curves of arbitrary degree
in the case of **segments**:

- when we sweep through a vertex, the \( y \)-order of the segments is reversed
- and hence the update of the \( Y \)-structure is fairly simple
- linear time in degree of the vertex

in the case of **curves**:

- when we sweep through a vertex, the \( y \)-order of the curves changes according to an arbitrary permutation
- or maybe not so arbitrary?
Sweeping Through a High Degree Vertex II

- assume for simplicity that common point is origin and (conceptually) write the curves as power series in $x$ and arrange in a trie.

\[y_1 = 2x + 3x^2 \quad y_2 = 2x + 1x^2 + 4x^3 \quad y_3 = 2x + 1x^2 \quad y_4 = 1x + 7x^2\]

- trie on left reflects order for small positive $x$
- order for small negative $x$ is obtained by flipping at odd depths
- only shape of trie is important
- paths for $y_2$ and $y_3$ split at depth 2 since mult of intersection is 3
- intersection = multipl 1, same slope = 2, same curvature = 3, …
- update time is linear in degree times cost of determining multiplicities of intersection
Boolean Ops on Polygons and Intersection of Curves

- once the arrangement induced by the boundaries is known, boolean ops reduce to graph traversal (see LEDAbook for details)
The Arrangement of a Single Conic

- a conic $C$: $\alpha_1 x^2 + \alpha_2 y^2 + 2\alpha_3 xy + 2\alpha_4 x + 2\alpha_5 y + \alpha_6 = 0$
- assume for simplicity that all conics are ellipses
- solve for $y$ and obtain equations for the upper and lower arc of $C$

$$C_{1,2}(x) = \frac{-b(x) \pm \sqrt{b(x)^2 - 4a(x)c(x)}}{2a(x)} \quad \ell_C(x) + \sqrt{q_C(x)}$$

where $a(x) = \alpha_2 \quad b(x) = 2\alpha_3 x + 2\alpha_5 \quad c(x) = \alpha_1 x^2 + 2\alpha_4 x + \alpha_6$.

- the $x$-coordinates of the start and endpoint of the arcs are given by the roots of $b(x)^2 - 4a(x)c(x) = 0$.

We can write them as one-root-expressions (OREs):

$$r + s\sqrt{t} \text{ with } r, s, t \in \mathbb{Q}$$
Curves of Arbitrary Degree

- Analysis of a single conic rests
  - on our complete understanding of conics
  - the ability to solve for $y$
  - the ability to write down explicite expressions for the $x$-coordinates of the endpoints of the arcs
- for arbitrary curves, we need to find the points of vertical tangent, the singularities, and the isolated points
  - all of them are intersections of $C$ and $C_y$ (the derivative of $C$ with respect to $y$)
  - needs more general techniques: see arrangements of two conics for a first flavor
The Arrangement of Two Conics I

- Two conics $C$ and $D$ can have up to four intersections.
- $x$-coordinates of intersections of $C$ and $D$ are roots of a polynomial $R_{CD}$ of degree at most four (see next slide).
- $R = R_{CD}$ is readily computed from $C$ and $D$.
- A degree four polynomial either has four distinct roots or roots are OREs.
- We can determine:
  - The number of real roots of $R$.
  - Their multiplicity.
  - Isolating intervals (if four distinct roots) or OREs.
- We use Descartes’ method.

$I = [a, b]$ is an isolating interval for a real root $x$ of $P$ iff $x$ is the unique root of $P$ in $I$. 
the equations for the arcs of $C$:

\[ \ell_C(x) + \sqrt{q_C(x)} \]

$\ell_C(x)$ is linear in $x$, $q_C(x)$ is quadratic in $x$

equating for $C$ and $D$ yields:

\[ \ell_C(x) + \sqrt{q_C(x)} = \ell_D(x) + \sqrt{q_D(x)} \]

isolate a root:

\[ \sqrt{q_C(x)} = \ell_D(x) - \ell_C(x) + \sqrt{q_D(x)} \]

square:

\[ q_C(x) = (\ell_D(x) - \ell_C(x))^2 + q_D(x) + 2(\ell_D(x) - \ell_C(x))\sqrt{q_D(x)} \]

isolate root:

\[ q_C(x) - (\ell_D(x) - \ell_C(x))^2 - q_D(x) = 2(\ell_D(x) - \ell_C(x))\sqrt{q_D(x)} \]

square:

\[ (q_C(x) - (\ell_D(x) - \ell_C(x))^2 - q_D(x))^2 = 4(\ell_D(x) - \ell_C(x))^2 q_D(x) \]

this is the desired equation $R_{CD}$
The Roots of a Degree Four Polynomial

- \( p(x) = x^4 + p_3x^3 + p_2x^2 + p_1x + p_0, p_i \in \mathbb{Q} \)
- \( p \) either has four distinct roots or the roots can be represented as OREs
- let \( g(x) := \gcd(p(x), p'(x)) \in \mathbb{Q}(x) \); the roots of \( g \) are the multiple roots of \( p \) with multiplicity reduced by one; let \( q(x) = p(x)/g(x) \in \mathbb{Q}(x) \).
- \( \deg(g) = 3, p \) has a fourfold root: \( p(x) = (x - \alpha)^4 \rightarrow \alpha = -p_3/4 \)
- \( \deg(g) = 2, g \)'s roots can be presented as OREs.
  - \( g \) has a double root, say \( \alpha \). Then \( p(x) = (x - \alpha)^3(x - \beta) \) and \( q(x) = (x - \alpha)(x - \beta) \). So \( \beta \) is also ORE.
  - \( g \) has two single roots, say \( \alpha \) and \( \beta \). Then \( p(x) = (x - \alpha)^2(x - \beta)^2 \).
- \( \deg(g) = 1, \) say \( g(x) = (x - \alpha) \) with \( \alpha \in \mathbb{Q} \). Then \( p(x) = (x - \alpha)^2(x - \beta)(x - \gamma) \) and \( (x - \beta)(x - \gamma) = p(x)/(x - \alpha)^2 \in \mathbb{Q}(x) \)
- \( \deg(g) = 0, p \) has four distinct roots
The Arrangement of Two Conics $C$ and $D$, Part II

- do arcs $C_i$ and $D_j$ intersect at ORE $x$?
- a conic $C$: $\alpha_1 x^2 + \alpha_2 y^2 + 2\alpha_3 xy + 2\alpha_4 x + 2\alpha_5 y + \alpha_6 = 0$
  
  $$C_{1,2}(x) = \frac{-b(x) \pm \sqrt{b(x)^2 - 4a(x)c(x)}}{2a(x)}$$

  where $a(x) = \alpha_2$, $b(x) = 2\alpha_3 x + 2\alpha_5$, $c(x) = \alpha_1 x^2 + 2\alpha_4 x + \alpha_6$.

- similarly for $D_{1,2}$

- is $C_i(x) = D_j(x)$? where $x = r + s\sqrt{t}$ with $r, s, t \in \mathbb{Q}$

- this question can be answered using CORE or LEDA reals, namely

  $$E = 0 \ ? \ where \ E = C_i(r + s\sqrt{t}) - D_j(r + s\sqrt{t})$$

- when written as a dag, $E$ contains three square roots; it has algebraic degree 8
Computing with Radical Expressions

Let $E$ be an expression with integer operands and operators $+,$ $-, \ast$ and $\sqrt{}$. Define

- $u(E) =$ value of $E$ after replacing $-$ by $+$.
- $k(E) =$ number of distinct square roots in $E$.

Then (BFMS, BFMSS)

$$E = 0 \quad \text{or} \quad |E| \geq \frac{1}{u(E)^{2^k(E)} - 1}$$

Theorem allows us to determine signs of algebraic expressions by numerical computation with precision $(2^{k(E)} - 1) \log u(E)$.

related work: Mignotte, Canny, Dube/Yap, Li/Yap, Scheinermann

extensions: division, higher-order roots, roots of univariate polynomials
Discussion I

How small can $A - B \sqrt{C}$ be, if non-zero? $A, B, C \in \mathbb{N}$.

$$|A - B \sqrt{C}| = \left| \frac{(A - B \sqrt{C})(A + B \sqrt{C})}{A + B \sqrt{C}} \right| = \frac{|A^2 - B^2 C|}{|A + B \sqrt{C}|} \geq \frac{1}{|A + B \sqrt{C}|} \geq \frac{1}{|A| + |B| \sqrt{C}}$$

This is a special case of the theorem

- $u(E) = |A| + |B| \sqrt{C}$
- $k = 1$
Discussion II

- Consider \( E = (\sqrt{x+1} + \sqrt{x}) \cdot (\sqrt{x+1} - \sqrt{x}) - 1 \) where \( x \) is an arbitrary integer.

- Observe \( E = 0 \).

- \( u(E) \approx 4x + 1 \approx 4x \) and \( k(E) = 2 \).

- Thus

\[
E = 0 \quad \text{or} \quad |E| \geq \frac{1}{u(E)^{2k(E)} - 1} \approx \frac{1}{(4x)^3}
\]

- It suffices to evaluate \( E \) with precision \( 3 \log(4x) = 3 \log x + 6 \).
Numerical Sign Computation for Algebraic Expressions

\[ \text{sep}(E) \leftarrow u(E)^{1-2^k(E)}; \quad \text{// bound from previous slide} \]
\[ k \leftarrow 1; \]
\[ \text{while (true)} \]
\[ \{ \text{compute an approximation } \tilde{E} \text{ with } |E - \tilde{E}| < 2^{-k}; \]
\[ \quad \text{if ( } |\tilde{E}| \geq 2^{-k} \text{ ) return } \text{sign}(\tilde{E}); \]
\[ \quad \text{if ( } 2^{-k} < \text{sep}(E)/2 \text{ ) return "zero"}; \quad \text{// since } |E| \leq 2^{-k} + 2^{-k} < \text{sep}(E) \]
\[ k \leftarrow 2 \cdot k; \quad \text{// double precision} \]
\]

- \( \tilde{E} \) is computed by numerical methods
- worst case complexity is determined by separation bound:
  maximal precision required is logarithm of separation bound
- easy cases are decided quickly (a big plus of the separation bound approach)
- strategy above is basis for sign test in LEDA \textit{reals}. 

Kurt Mehlhorn, MPI for Informatics and Saarland University

Conic Polygons – p.21/32
The Arrangement of Two Conics $C$ and $D$, Part III

- have a deg four polynomial $R = R_{CD}$ and a root $x$ of $R$
- do $C_i$ and $D_j$ intersect at $x$?
  - if $x$ is given as ORE, compare $C_i(x)$ and $D_j(x)$ using LEDA-reals
  - if $x$ is not given as an ORE, $x$ is a simple root of $R$ and hence
    - we have an isolating interval $[a,b]$ for $x$
    - $C_i$ and $D_j$ intersect at most once in $[a,b]$ and, if so, at $x$
    - if $C_i$ and $D_j$ intersect at $x$, they cross at $x$
    - cross if $\text{sign}(C_i(a) - D_j(a)) \neq \text{sign}(C_i(b) - D_j(b))$
    - decide the latter using LEDA reals
The Arrangement of Two Conics $C$ and $D$, Part IV

- assume $C_i$ and $D_j$ intersect at $x$.
- what is multiplicity $c$ of intersection?
- $m = \text{multiplicity of } x \text{ as root of } R = R_{CD}$; then $m \leq 4$ and $c \leq m$.
- if $m = 1$ we have $c = 1$ and are done
- if $m \geq 2$, we have an ORE for $x$
- compute tangent vectors for $C_i$ and $D_j$ at $x$ as LEDA-reals by considering partial derivatives
- if not parallel, $c = 1$, otherwise $c \geq 2$.
- $d = \text{multiplicity of intersection of } C_{1-i} \text{ and } D_{1-j} \text{ at } x$.
- then $c + d = m$ this requires a little lemma
- determine whether $d = 0$, $d = 1$, $d \geq 2$.
- together with $c \geq 2$ and $m \leq 4$, this determines $c$.
The Other Predicates

Use Similar Techniques
Correctness of Implementation

- have manually checked output for a small number of examples
- have run alg on a large number of random examples
  - always checked that arrangement computed by the sweep is a planar map, i.e., satisfies Euler’s formula

\[
\text{number of vertices} - \text{number of edges} + \text{number of face cycles} = 2
\]

- for boolean ops on polygons \( P \) and \( Q \):
  threw random points \( p \) into the plane and checked

\[
(p \in P) \text{ op } (p \in Q) = p \in (P \text{ op } Q)
\]
• $s_1 \cap s_2$ is recomputed and reinserted into X-struct after $s_3$ is swept
• this amounts to a very expensive equality test (outcome is equal)
• alternative I
  • store intersection in a dictionary (under key $(s_1, s_2)$)
  • optimization was already used for line segments, but now it really pays
• alternative II
  • store construction history of point (and not just coordinates)
  • test for identity instead of equality
  • combinatorial test instead of numerical test
Optimization II: Retain Geometric Information

- default implementation of segments: store endpoints
- when endpoints are circle points it is very costly to recover the underlying line
- better implementation: store underlying line and the endpoints
- generalizes to arbitrary curves: store underlying curve and two curve points
Optimization III: Number Types

- always use the simplest possible number type
- integers, rationals, radical expressions, general algebraic numbers
- use floating point filtering
Optimization IV: Exploit Degeneracies

• degeneracies frequently lead to simpler coordinates

• an example

• if three rational circles (rational center plus rational squared radius) have a point in common, then
  • either this point is rational
  • or the three centers are collinear (and hence they have two points in common)

• a general fact: if the singularity of highest order is unique, it has rational coordinates
Optimizations for Circle Arcs and Line Segments

- no optimizations
- using structural filtering
- using rational numbers

Time (sec.)

Number of original segments

Optimizations and careful implementations of predicates are the key

Kurt Mehlhorn, MPI for Informatics and Saarland University
Timings II

Line segments vs. Circular arcs

LED A line segment sweep
line segments

circular arcs

there is still a long way to go

Kurt Mehlhorn, MPI for Informatics and Saarland University
Beyond Conics

- computational geometry part works for arbitrary curves
- but the key to success are the constructions and predicates
- they require considerable refinement, e.g.,
  - roots of resultant cannot be represented explicitly, but only by isolating intervals
  - curves of degree three or more may have singularities
  - two events at the same $x$-coordinate are a headache
  - analysis of a curve $f(x,y)$ at $x$-coordinate $\alpha$ requires to determine the roots of $f(\alpha,y)$. In general, $\alpha$ is an algebraic number.
- see home pages of Eric Berberich, Arno Eigenwillig, Michael Hemmer, Michael Kerber, Kurt Mehlhorn, Michael Sagraloof, and Nicola Wolpert for more recent work