# Visualization of algebraic curves.

L. Alberti, B. Mourrain,
GALAAD INRIA Sophia Antipolis

November 2, 2007

# Topology of a planar curve

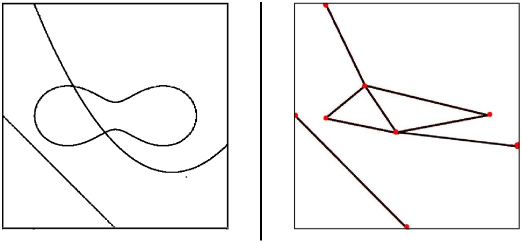
☞What do we have ?

❒ $f(x,y) \in \mathbb{Q}[x,y]$ square-free, defining a curve $\mathcal{C}$.

❒ A rectangular box $\mathcal{D} = [a,b] \times [c,d]$.

☞What do we want ?

❒ A planar graph (points and segments) isotopic to $\mathcal{C}$ in $\mathcal{D}$.



# Motivation

☞Intended for mathematicians as a research tool.

☞Computer Assisted Design:

- ❒ Visualization of features.
- ❒ Computations on parametrized surfaces.

☞Geometric optic analysis, caustics.

# Features and strong points

☞Certified output (unlike marching cube or spray points methods).

☞Subdivision method $\Longrightarrow$ complexity related to geometry.

☞Avoids computation with exact numbers $\Longrightarrow$ Fast.

☞New approach to determining the topology of smooth curves.

☞New approach to determining the topology near singular points.

# Example

☞Works on high degree polynomials with big coefficients.

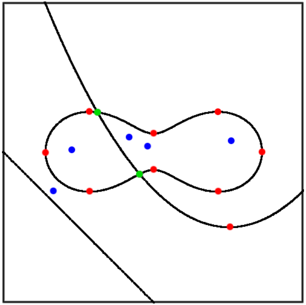
☞Example of a self-intersection curve of a quartic surface of total degree 50 :

```
290171273218648638010414503766589378194920450470147559546420395798997196147687839420852522870126610736768664\
    2416364693366201324858842952344545987747498583537668781910332000 x^3 y^29 - 1287137987422010199437099556591\
    59185267768378376454402610649374313531545065904586465805385813932393399449049405653968633751444386223251\
    842230166289913146211851911164871044761600 x^24 y^7 + 27630588966669286628073896943237236264708445261217111\
    18455982452396471189352869134255784278274619085389214420753431963914174682353894771544952687555306122921\
    8687253401600000 x^38 y^16 + 690930247274108007091721250714768003289001734839308348738957858963131058897930\
    023331249926566142870065085227691743223651011498272269485780633853856204999536163269427200000 x^38 y^20 +
    55753153611730609496660368036051267199463778788399335981536910193178660588694903161364093061120987119724\
    06089088858225553561466178435507321280335460201664185979394990080000 x^38 y^18 - 54254410132027664522912787\
    59213714331313736143218273051141429582603702195974076589868227143607654643322198116744551905032251647751\
    6868964586893280417570050280146440396800000 x^38 y^15 - 220957650263775280794465072925604718519846915651025\
    93128753907669398989900560624959535449082610376700738640177494115758264676488106972544718351900657655977\
    58032232356932480 x^3 y^30 + 419049997758227420511287291121464865696822287093283823102964073115873081477960\
    267410570758462718185467208353142882820932863455272335095315165506590718604746074315865423547033600 x^28
    y^18 • • • • • •
```

# Subdivision method

Interesting points:

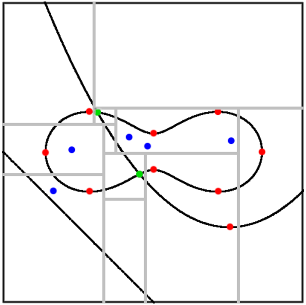
- $x$-critical points of $\mathcal{C}$: $f(x,y) = \partial_y f(x,y) = 0$.
  $y$-critical points of $\mathcal{C}$: $f(x,y) = \partial_x f(x,y) = 0$.
- Singular points of $\mathcal{C}$: $f(x,y) = \partial_x f(x,y) = \partial_y f(x,y) = 0$.
- Extremal points of $f$: $\partial_x f(x,y) = \partial_y f(x,y) = 0$.



# Subdividing the box

Types of boxes:

- $x$-regular box: Contains no $x$-critical points.
  $y$-regular box: Contains no $y$-critical points.
- Simple singular box: Contains exactly one extremal point and this point is on $\mathcal{C}$.



# Isolating the points

☞ Use of Bernstein basis and sleeve methods

- ❒ Univariate solving
- ❒ Multivariate solving in non degenerate cases
- ❒ Preconditionning and floating point computations $\Longrightarrow$ Fast
- ❒ Certified computation (Separation bounds)

# Isolating the points

☞Use of Bernstein basis and sleeve methods

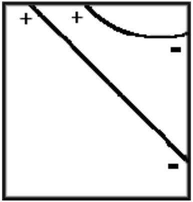
- ❒ Univariate solving
- ❒ Multivariate solving in non degenerate cases
- ❒ Preconditionning and floating point computations $\Longrightarrow$ Fast
- ❒ Certified computation (Separation bounds)

☞Univariate Rational Representation and interval arithmetic

- ❒ Reduces multivariate case to univariate case
- ❒ Handles degenerate cases

# Handling the regular boxes

We can assume the box to be $x$-regular wlg.
☞The "$x$-index" at a point of $\mathcal{C}$ on the boundary of the box is positive if the branch goes to the right and negative otherwise.

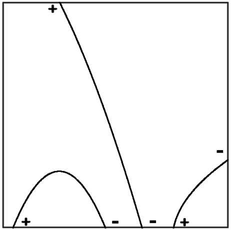



**$x$-index lemma**

If $a$ and $b$ are connected by a branch of $\mathcal{C}$ in the box then one has positive $x$-index and the other negative $x$-index.
The one with positive $x$-index has a lower absciss than the one with negative $x$-index.

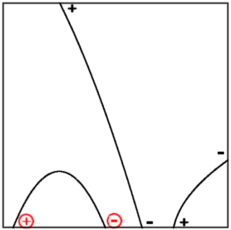
## Connection algorithm

- Choose $b$ with negative $x$-index and minimal absciss.
- Choose $a$ its neighbor on the boundary that has positive $x$-index and absciss lower than $b$.
- Connect $a$ and $b$.
- Iterate from the first step as many times as necessary.



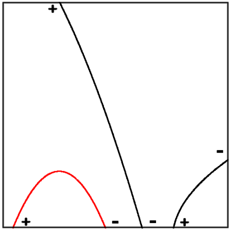
## Connection algorithm

- Choose $b$ with negative $x$-index and minimal absciss.
- Choose $a$ its neighbor on the boundary that has positive $x$-index and absciss lower than $b$.
- Connect $a$ and $b$.
- Iterate from the first step as many times as necessary.



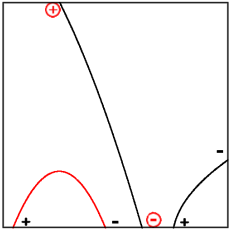
## Connection algorithm

- Choose $b$ with negative $x$-index and minimal absciss.
- Choose $a$ its neighbor on the boundary that has positive $x$-index and absciss lower than $b$.
- Connect $a$ and $b$.
- Iterate from the first step as many times as necessary.



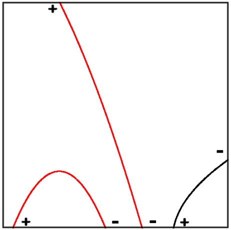
## Connection algorithm

- Choose $b$ with negative $x$-index and minimal absciss.
- Choose $a$ its neighbor on the boundary that has positive $x$-index and absciss lower than $b$.
- Connect $a$ and $b$.
- Iterate from the first step as many times as necessary.



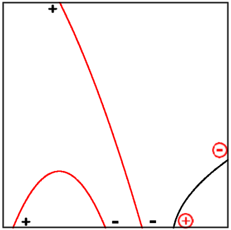
## Connection algorithm

- Choose $b$ with negative $x$-index and minimal absciss.
- Choose $a$ its neighbor on the boundary that has positive $x$-index and absciss lower than $b$.
- Connect $a$ and $b$.
- Iterate from the first step as many times as necessary.



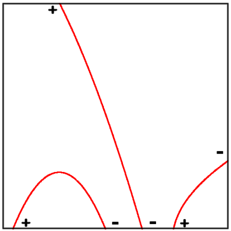
## Connection algorithm

- Choose $b$ with negative $x$-index and minimal absciss.
- Choose $a$ its neighbor on the boundary that has positive $x$-index and absciss lower than $b$.
- Connect $a$ and $b$.
- Iterate from the first step as many times as necessary.



## Connection algorithm

- Choose $b$ with negative $x$-index and minimal absciss.
- Choose $a$ its neighbor on the boundary that has positive $x$-index and absciss lower than $b$.
- Connect $a$ and $b$.
- Iterate from the first step as many times as necessary.

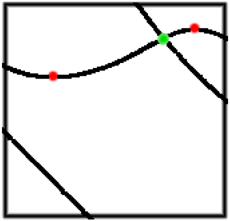


# Handling the simple singular boxes

**Local conic structure theorem**

For a sufficiently small box $\mathcal{D}$ containing a singular point, the topology of $\mathcal{C}$ in the box is a cone over $\mathcal{C} \cap \partial\mathcal{D}$.

☞We don't know how small the box has to be.



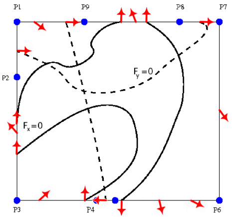
☞Criterion : The box is simple singular and the number of half-branches at the singular point is the number of points on the boundary.

# Counting the number of half-branches

## Definition of the topological degree

Let $F : \mathbb{R}^2 \to \mathbb{R}^2$ such that $F(x,y) \neq 0$ on the boundary of the box $\mathcal{D}$. Let $p_i$ clockwise-ordered points on the boundary so that $F_x$ or $F_y$ has constant sign between two of them. Let $\sigma_i = 1$ if $F_x$ as constant sign between $p_i$ and $p_{i+1}$, 0 otherwise.
The topological degree of $F$ in the box is :

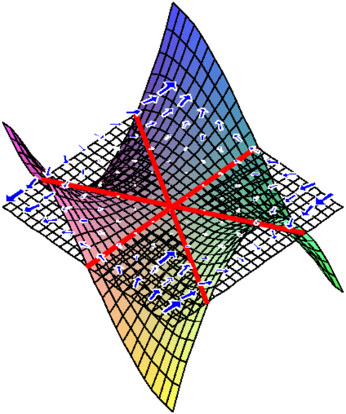
$$\deg(F,\mathcal{D}) = \frac{1}{8}\sum_{i=1}^{g}(-1)^{\sigma_i - 1}\begin{vmatrix} \mathrm{sg}(f_{\sigma_i}(p_i)) & \mathrm{sg}(f_{\sigma_i}(p_{i+1})) \\ \mathrm{sg}(f_{\sigma_i+1}(p_i)) & \mathrm{sg}(f_{\sigma_i+1}(p_{i+1})) \end{vmatrix}$$



## Khimshiashvili's theorem

When there is no other extremal point of $f$ in a box $\mathcal{D}$, the number of half-branches at the singular point is

$$2(1 - \deg(\mathrm{grad}\ f, \mathcal{D}))$$



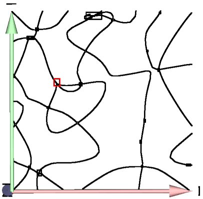
# The final algorithm

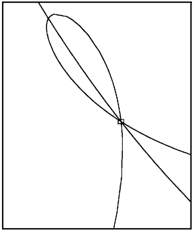
## Topology computation algorithm

- Isolate each singular point from the other extremal points.
- For each simple singular box
  - Test if the number of half-branches is the same as the number of points on the boundary.
  - If yes, keep this box and move on to the next simple singular box.
    If not, refine the isolation box and test until the test succeeds.
- Isolate the $x$-critical points from the $y$-critical points and the isolating boxes for the singular points.
- Partition the rest of the space in boxes (that are thus $x$-regular).
- Apply the relevant connection algorithm in each box.

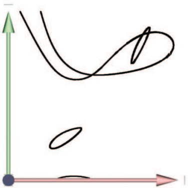
# Worked examples

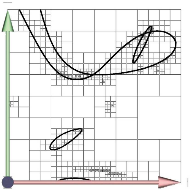
Ridge curve of a parametric surface in CAO:





A polynomial with degree 50 obtained as the projection of the self-intersection locus of a quartic surface:





## Contributors:

- Chen Liang
- B. Mourrain
- J.P. Pavone
- J. Wintz

## Tools:

- SYNAPS: a C++ library for symbolic and numeric computations. GPL, http://synaps.inria.fr
- AXEL, an algebraic-geometric modeler, GPL http://axel.inria.fr

## Acknowledgements:

AIM@Shape (IST Network of Excellence 506766)
ACS (IST FET Open 006413).