Workshop of Geometry Processing and Applications

Interval Methods in Computer Graphics

Luiz Henrique de Figueiredo (IMPA)

September 2004

Motivation

- How do I plot an implicit curve?
  - ◇ Must solve $f(x, y) = 0$
  - ◇ Solution is a curve, but where is it?

- How do I render an implicit surface?
  - ◇ Must solve $f(x, y, z) = 0$ for (x, y, z) on a ray
  - ◇ Solution is one or more points, but need point closest to eye!

- How do I intersect two parametric surfaces?
  - ◇ Must solve $f(u, v) = g(s, t)$
  - ◇ Solution is a set of curves in space and a set of curves in each parametric plane. Where are they? How do they match?

Motivation

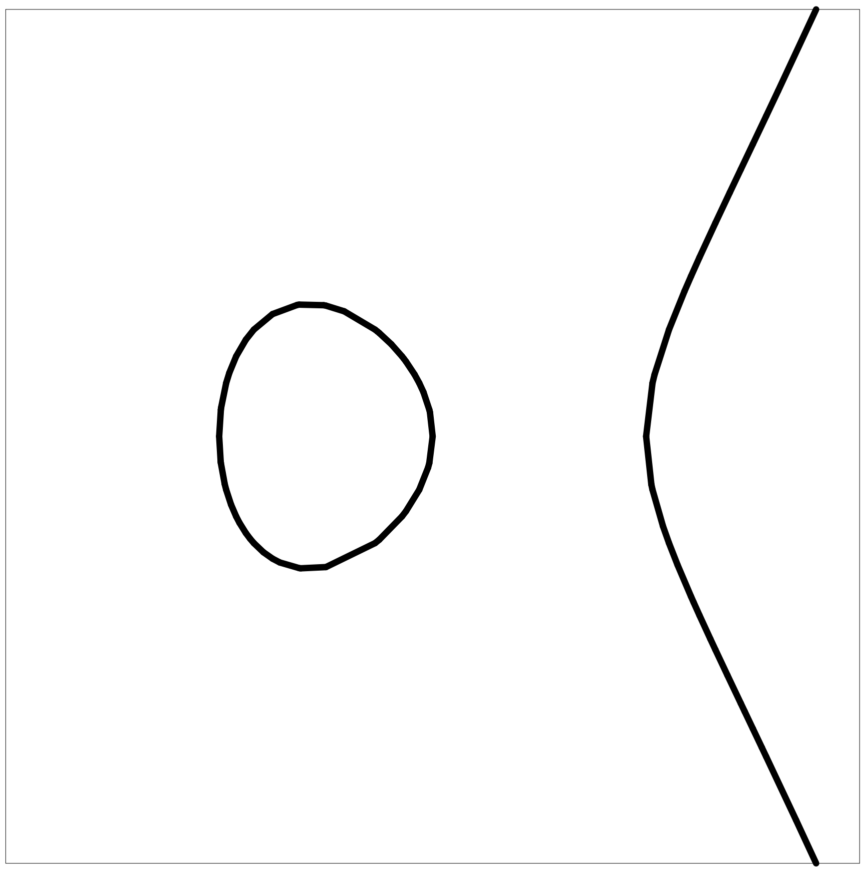
- How do I plot an implicit curve?
  - ◇ Must solve $f(x, y) = 0$
  - ◇ Solution is a curve, but where is it?

- How do I render an implicit surface?
  - ◇ Must solve $f(x, y, z) = 0$ for (x, y, z) on a ray
  - ◇ Solution is one or more points, but need point closest to eye!

- How do I intersect two parametric surfaces?
  - ◇ Must solve $f(u, v) = g(s, t)$
  - ◇ Solution is a set of curves in space and a set of curves in each parametric plane. Where are they? How do they match?

Plotting an implicit curve

$y^2 - x^3 + x = 0$ $\Omega = [-2, 2] \times [-2, 2]$

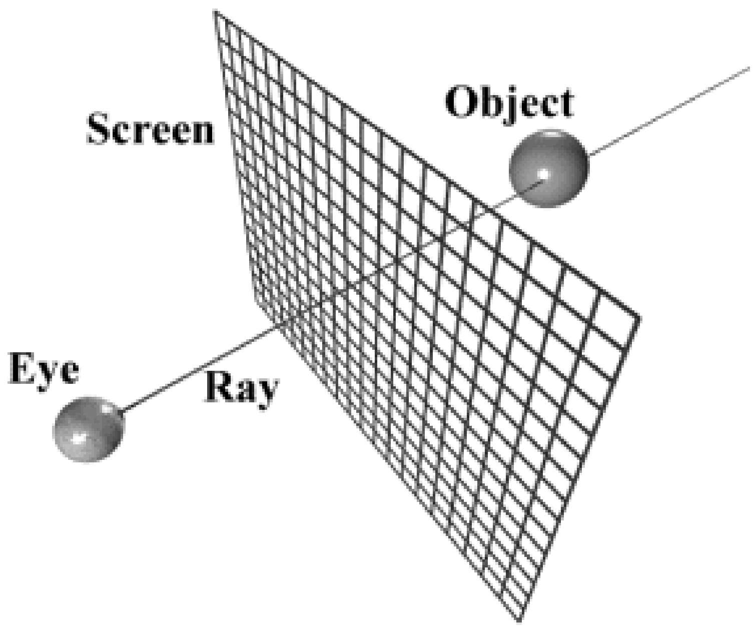


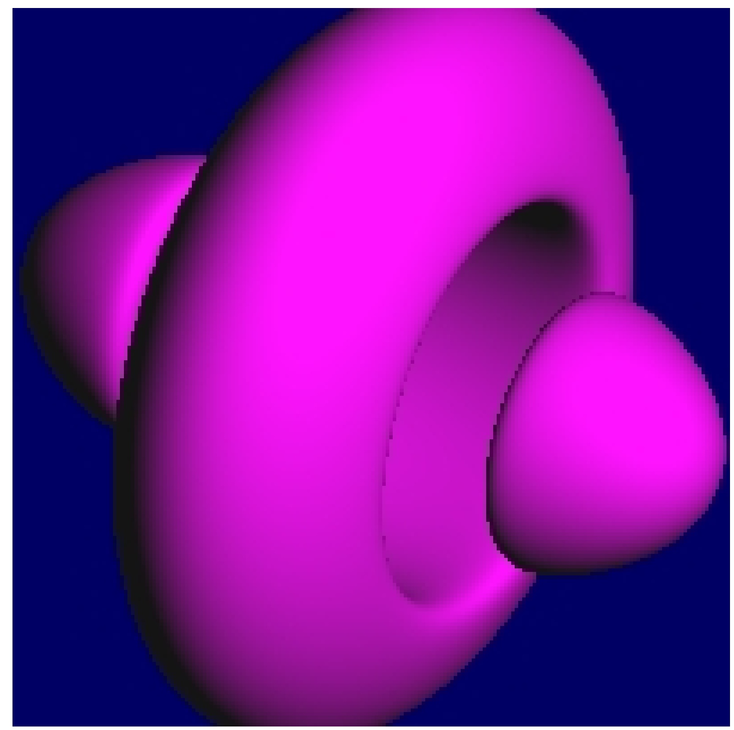
Motivation

- How do I plot an implicit curve?
  - ◇ Must solve $f(x, y) = 0$
  - ◇ Solution is a curve, but where is it?
- How do I render an implicit surface?
  - ◇ Must solve $f(x, y, z) = 0$ for (x, y, z) on a ray
  - ◇ Solution is one or more points, but need point closest to eye!
- How do I intersect two parametric surfaces?
  - ◇ Must solve $f(u, v) = g(s, t)$
  - ◇ Solution is a set of curves in space and a set of curves in each parametric plane. Where are they? How do they match?

Rendering implicit surfaces

$$4(x^4 + (y^2 + z^2)^2) + 17x^2(y^2 + z^2) - 20(x^2 + y^2 + z^2) + 17 = 0$$





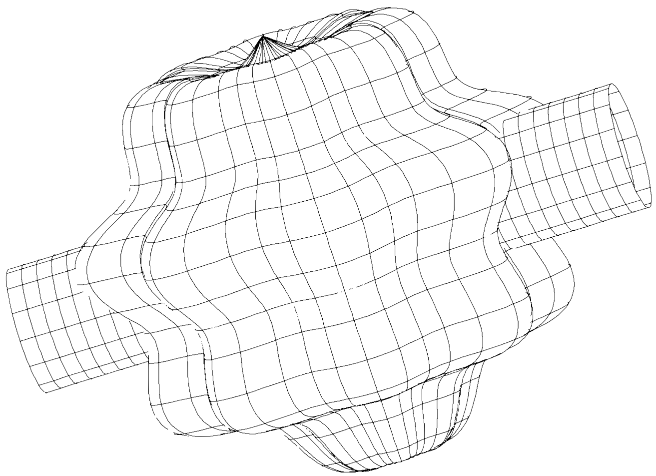
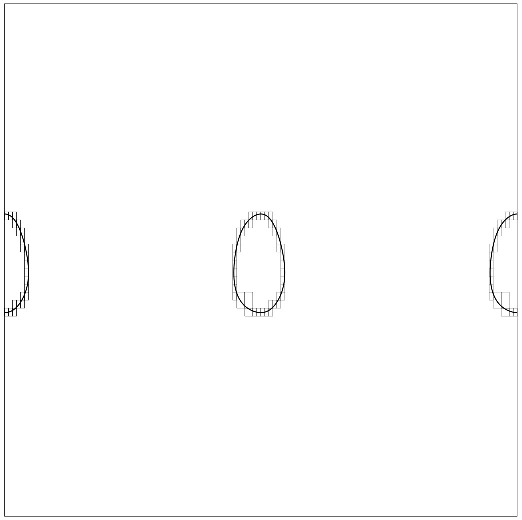
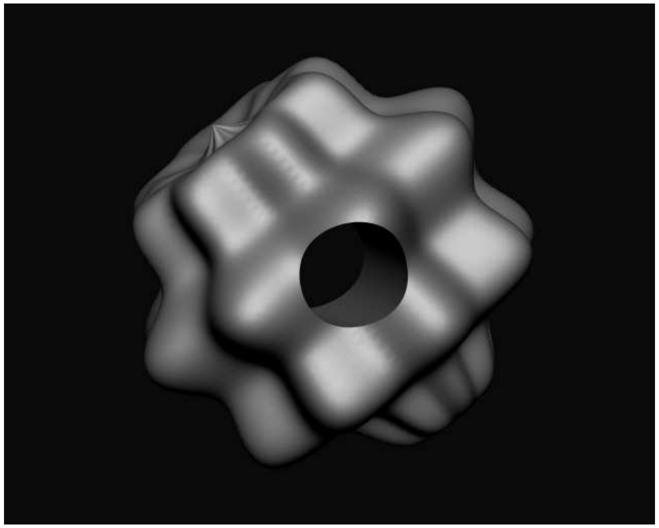
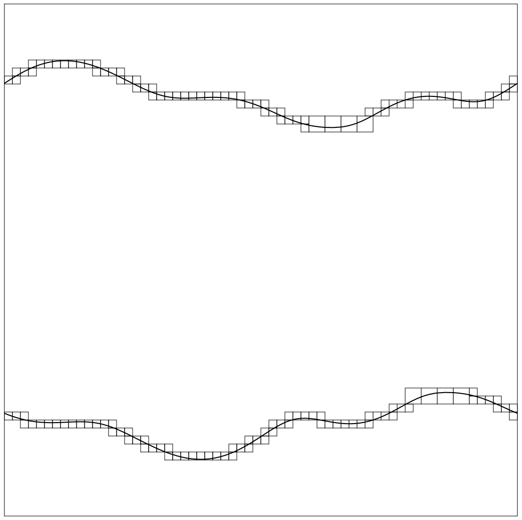
Motivation

- How do I plot an implicit curve?
  - ◇ Must solve $f(x, y) = 0$
  - ◇ Solution is a curve, but where is it?

- How do I render an implicit surface?
  - ◇ Must solve $f(x, y, z) = 0$ for (x, y, z) on a ray
  - ◇ Solution is one or more points, but need point closest to eye!

- How do I intersect two parametric surfaces?
  - ◇ Must solve $f(u, v) = g(s, t)$
  - ◇ Solution is a set of curves in space and a set of curves in each parametric plane. Where are they? How do they match?

Intersecting two parametric surfaces

(Snyder, 1992)

Interval arithmetic

Can we trust floating-point arithmetic?

Rump's example – Evaluate this innocent-looking polynomial expression:

$$f = 333.75y^6 + x^2(11x^2y^2 - y^6 - 121y^4 - 2) + 5.5y^8 + x/(2y),$$

for $x = 77617$ and $y = 33096$.

```
f:=333.75*y^6+x^2*(11*x^2*y^2-y^6-121*y^4-2)+5.5*y^8+x/(2*y);
                                f := 1.172603940

f:=33375/100*y^6+x^2*(11*x^2*y^2-y^6-121*y^4-2)+55/10*y^8+x/(2*y);
                                       54767
                                f := - -----
                                       66192
evalf(f,10);
                                -0.8273960599
```

Not Maple's fault! Running gcc under Linux gives 5.76461×10^{17}.

Culprit is catastrophic cancellation of floating-point arithmetic!

Interval arithmetic

- To improve reliability of floating-point computations (Moore, 1960)
- Represent quantities as intervals:

$$x \sim [a, b] \Rightarrow x \in [a, b]$$

- Operate with intervals generating other intervals:
  - ◇ Simple formulas for elementary operations and functions:

$$\begin{aligned}
[a,b] + [c,d] &= [a+c, b+d] \\
[a,b] \times [c,d] &= [\mathrm{min}\{ac, ad, bc, bd\}, \mathrm{max}\{ac, ad, bc, bd\}] \\
[a,b] \,/\, [c,d] &= [a,b] \times [1/d, 1/c] \\
[a,b]^2 &= [0, \mathrm{max}(a^2, b^2)] \ \text{ when } \ 0 \in [a,b] \\
\exp\,[a,b] &= [\exp(a), \exp(b)] \\
&\cdots
\end{aligned}$$

  - ◇ Automatic extensions for complicated expressions
  - ◇ Rounding control available in modern floating-point units (IEEE 754)

Interval arithmetic

- Every expression f has an interval extension F :

$$x_i \in X_i \Rightarrow f(x_1, \ldots, x_n) \in F(X_1, \ldots, X_n)$$

- Interval computations not immune to roundoff errors
  Wide results alert user of catastrophic cancellation
- Roundoff errors are not our main motivation!
- Interval computations allow range estimates and avoid point sampling

$$F(X) \supseteq f(X) = \{f(x) : x \in X\}$$

For instance

$$\begin{aligned} 0 \notin F(X) &\Rightarrow 0 \notin f(X) \\ &\Rightarrow f = 0 \text{ has no solution in } X \end{aligned}$$

This is a computational proof!

Interval probing of implicit curve

$$y^2 - x^3 + x = 0$$

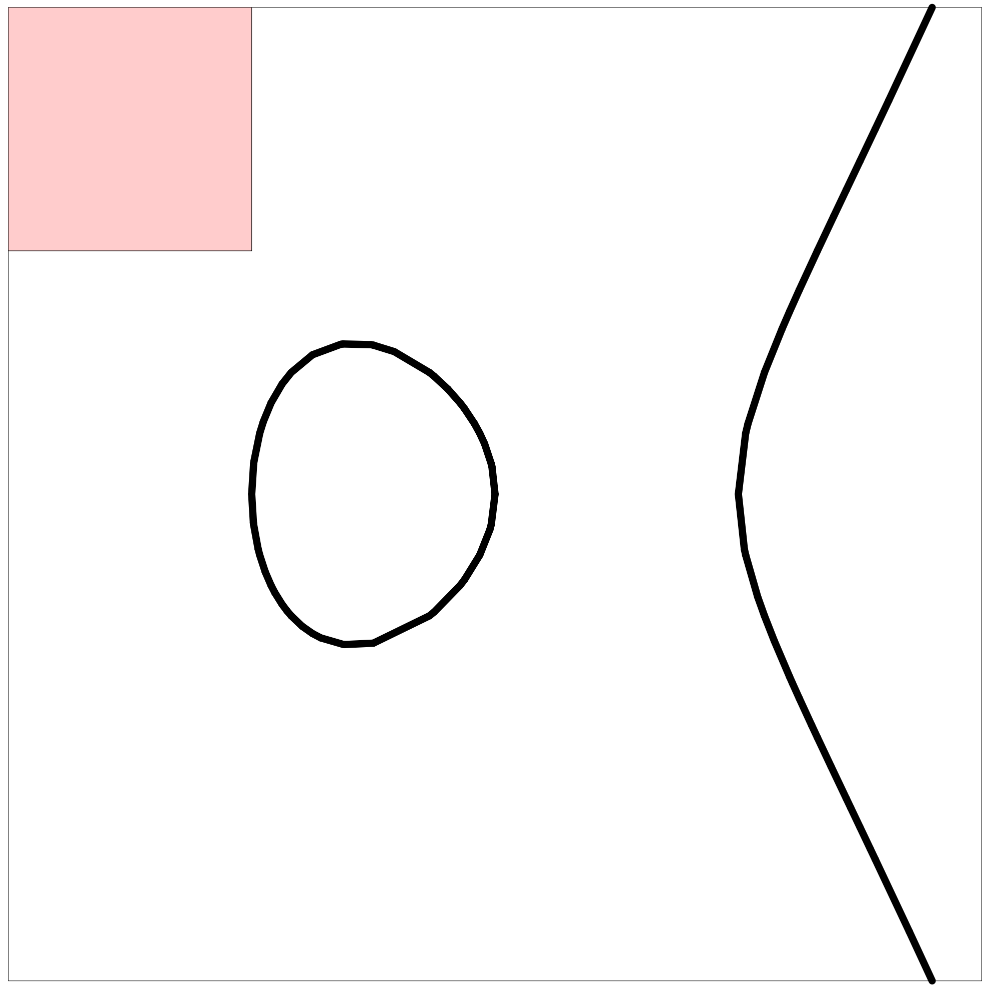
$$\begin{aligned} X &= [-2, -1] \\ Y &= [1, 2] \\ X^3 &= [-8, -1] \\ -X^3 &= [1, 8] \\ -X^3 + X &= [-1, 7] \\ Y^2 &= [1, 4] \\ Y^2 - X^3 + X &= [0, 11] \end{aligned}$$

- Interval estimates not tight

$$f(X, Y) = [1, 10] \subset [0, 11]$$

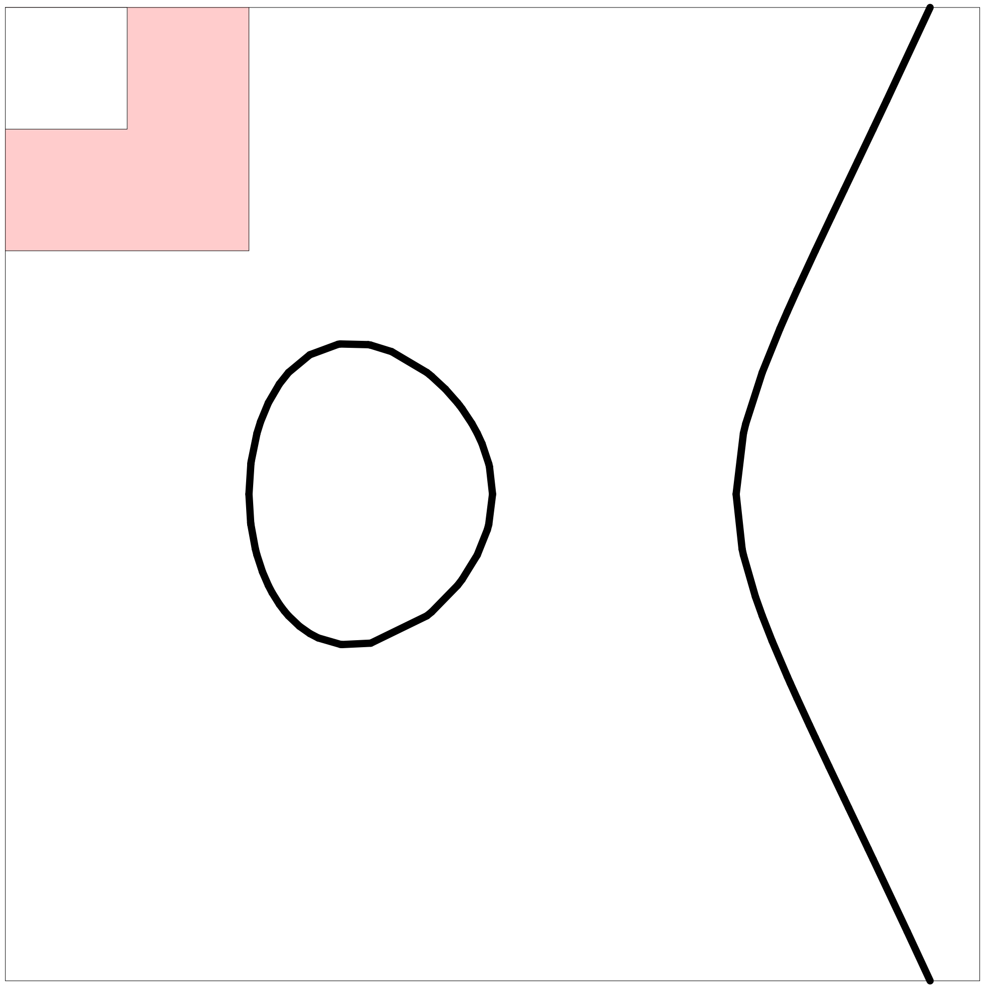
- Interval estimates improve as intervals shrink

Interval probing of implicit curve



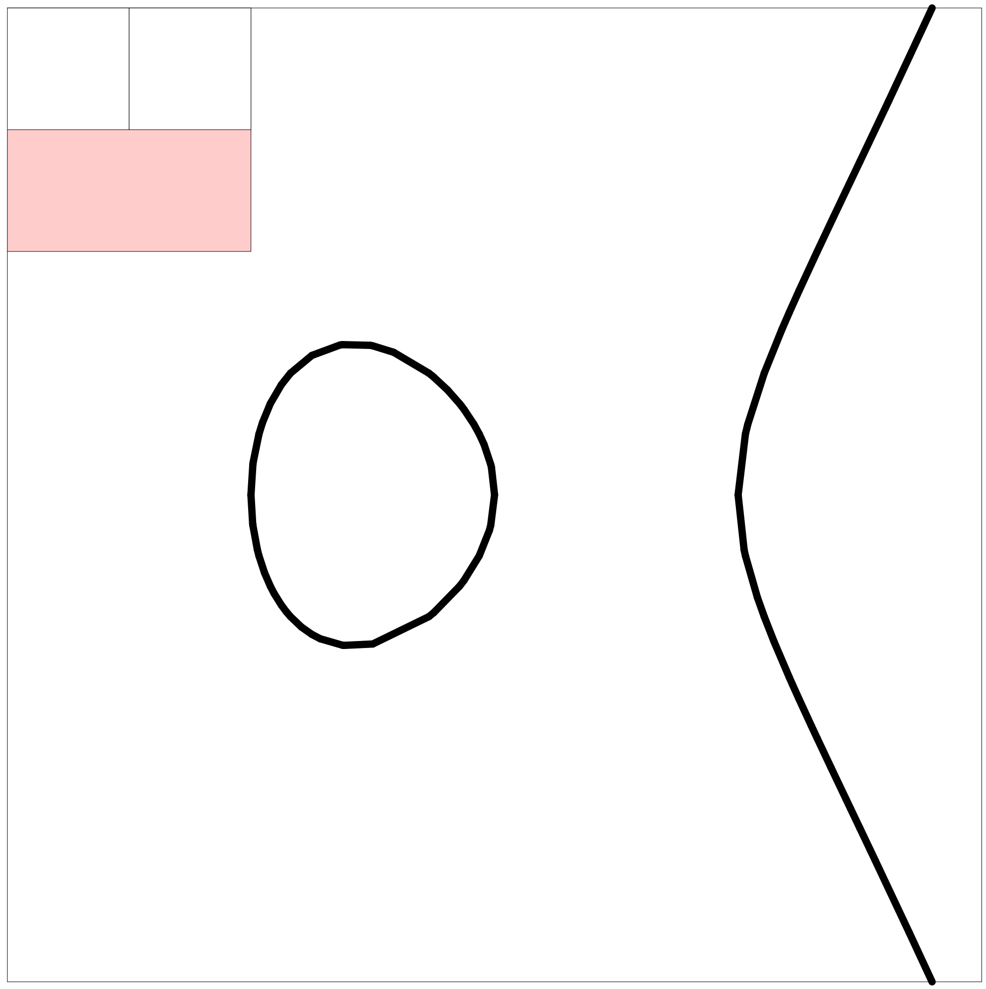
$[-2, -1] \times [1, 2]$ $[0, 11]$ yes?

Interval probing of implicit curve



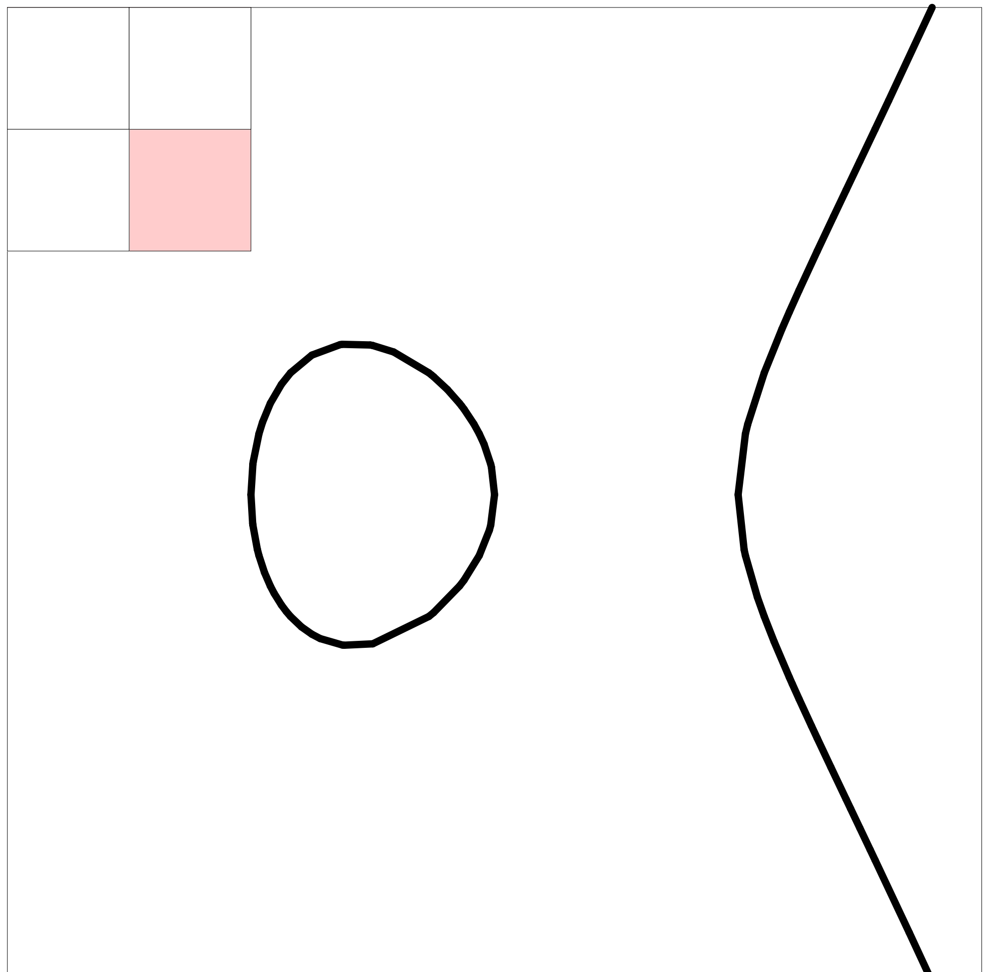
$[-2, -1.5] \times [1.5, 2]$ $[3.625, 10.5]$ no

Interval probing of implicit curve



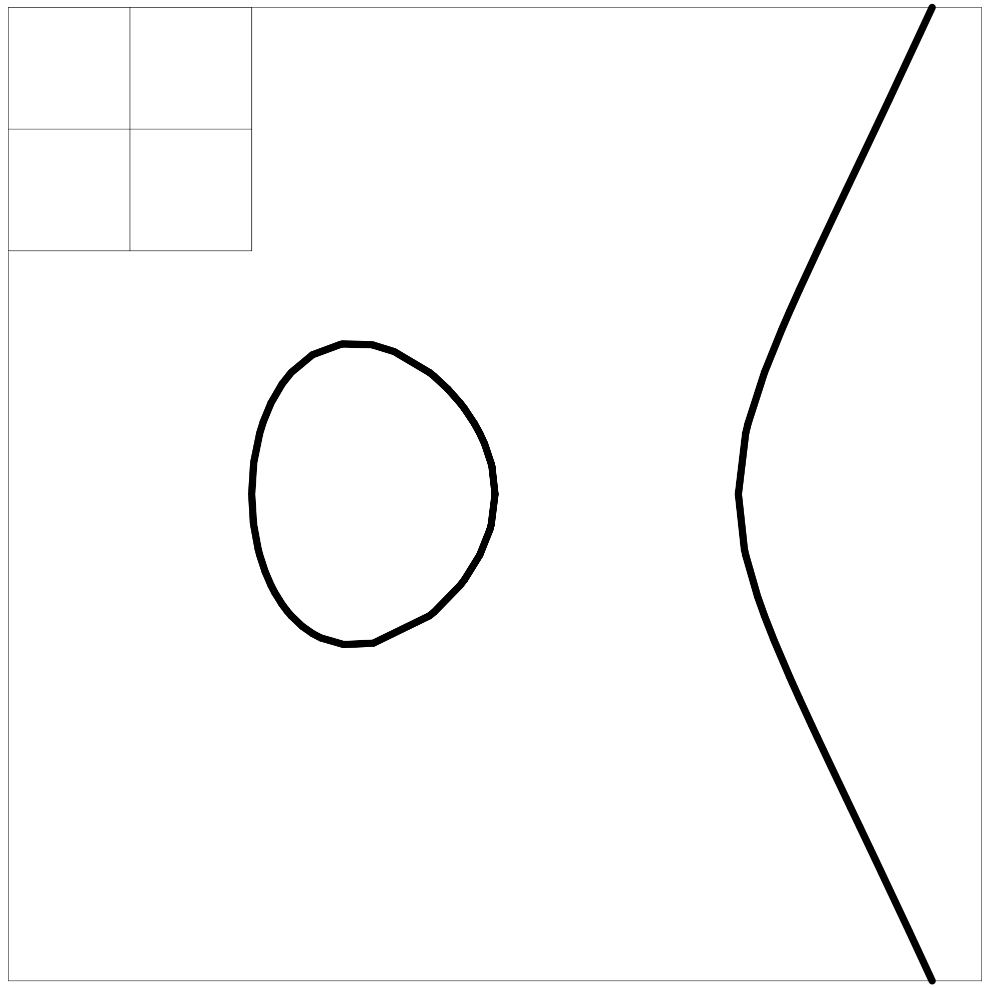
$[-1.5, -1] \times [1.5, 2]$ $[1.75, 6.375]$ no

Interval probing of implicit curve



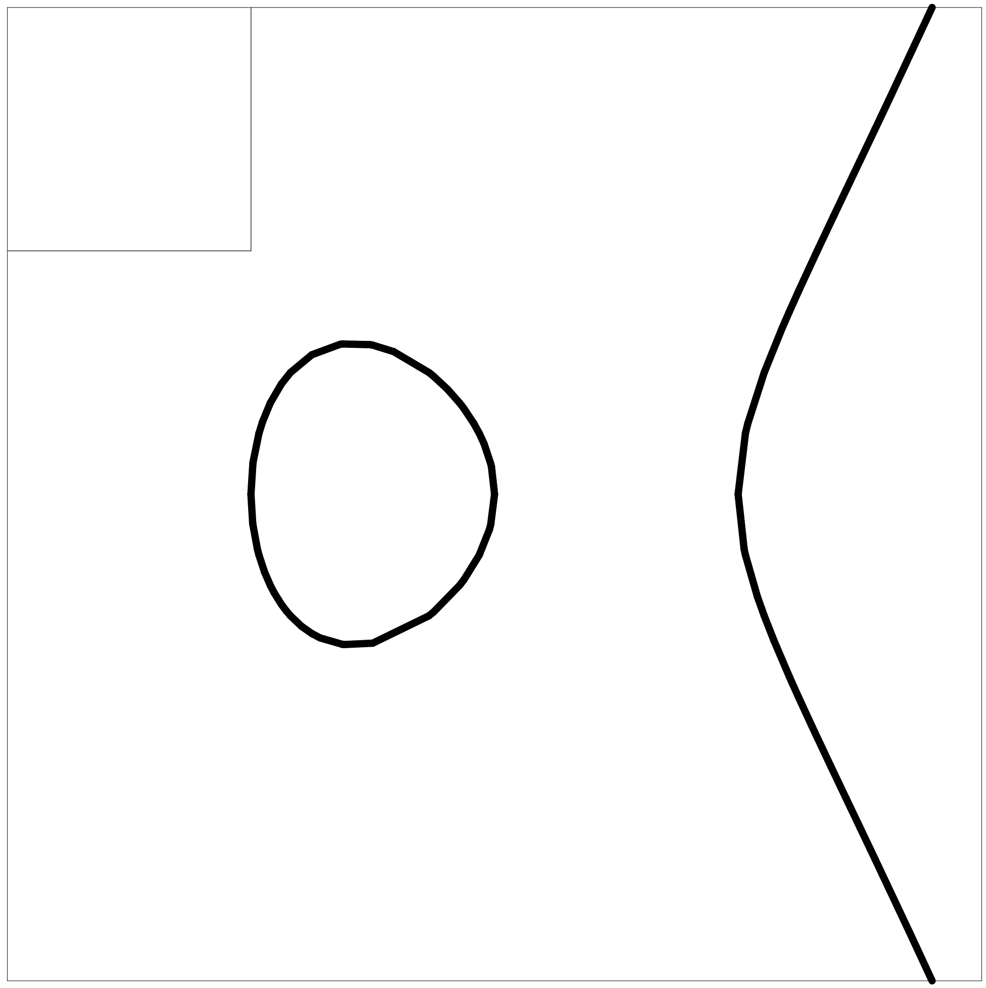
$[-2, -1.5] \times [1, 1.5]$ $[2.375, 8.75]$ no

Interval probing of implicit curve



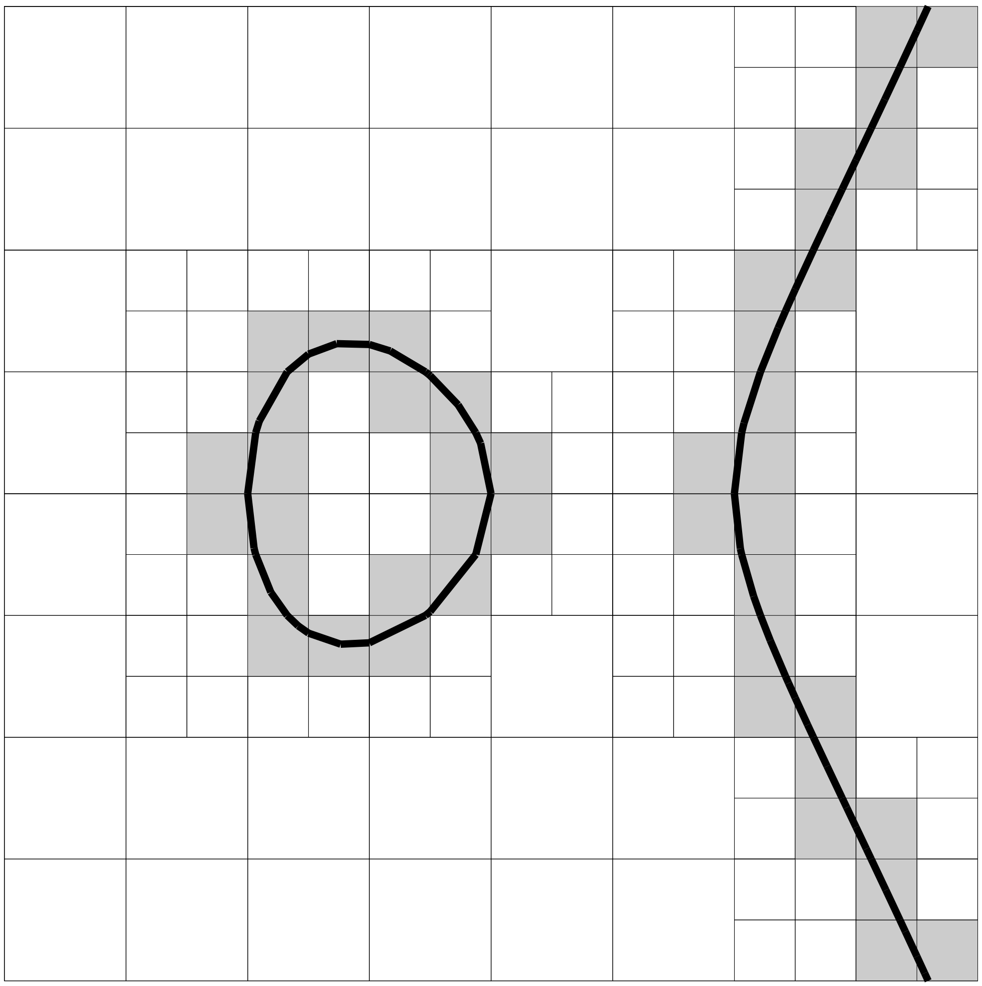
$[-1.5, -1] \times [1, 1.5]$ $[0.5, 4.625]$ no

Interval probing of implicit curve



$[-2, -1] \times [1, 2]$ $[0.5, 10.5]$ no!

Approximation of implicit curve



Robust adaptive enumeration

- Recursive exploration of domain Ω starts with explore(Ω)
- Discard subregions X of Ω when $0 \notin F(X)$
  = proof that X does not contain any part of the curve!

```
explore(X):
    if 0 ∉ F(X) then
        discard X
    elseif diam(X) < ε then
        output X
    else
        divide X into smaller pieces X_i
        for each i, explore(X_i)
```

- Output cells have the same size: only spatial adaption

Suffern–Fackerell (1991), Snyder (1992)

Robust adaptive approximation

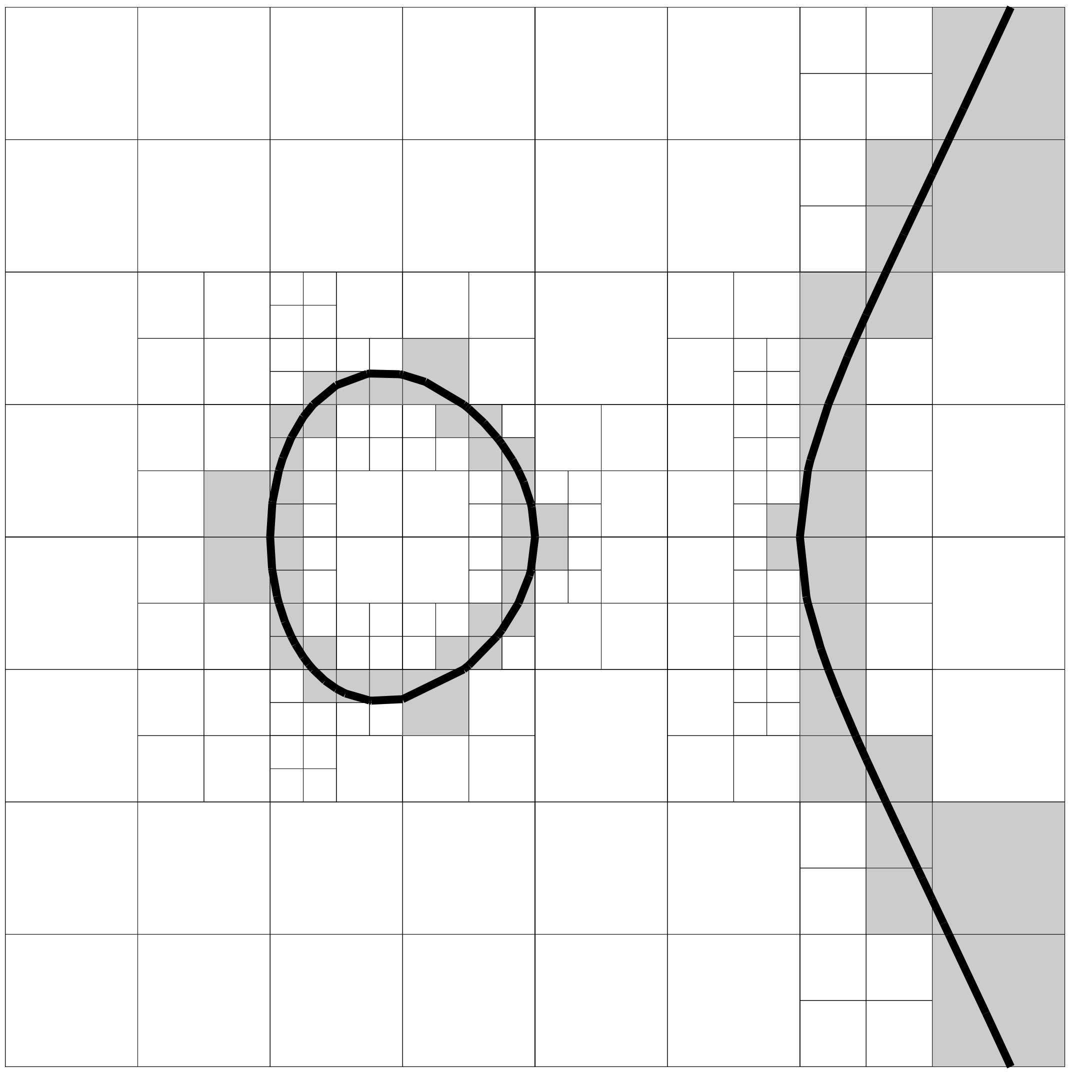
- Estimate curvature by gradient variation
- $G =$ inclusion function for the normalized gradient of f
- $G(X)$ small $\Rightarrow$ curve approximately flat inside X

```
explore(X):
    if 0 ∉ F(X) then
        discard X
    elseif diam(X) < ε or diam(G(X)) < δ then
        approx(X)
    else
        divide X into smaller pieces X_i
        for each i, explore(X_i)
```

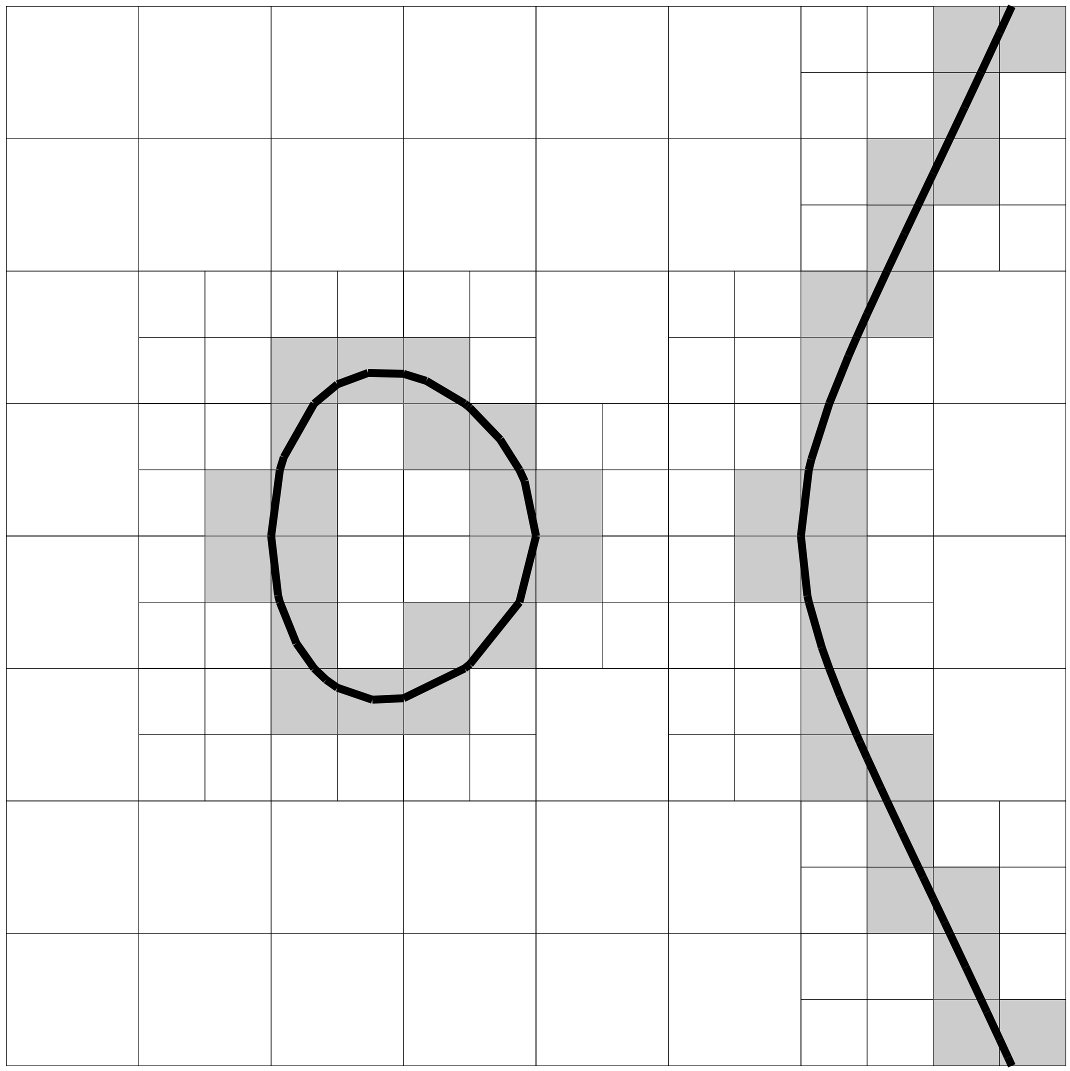
- Output cells vary in size: spatial and geometrical adaption

Lopes–Oliveira–Figueiredo (2002)

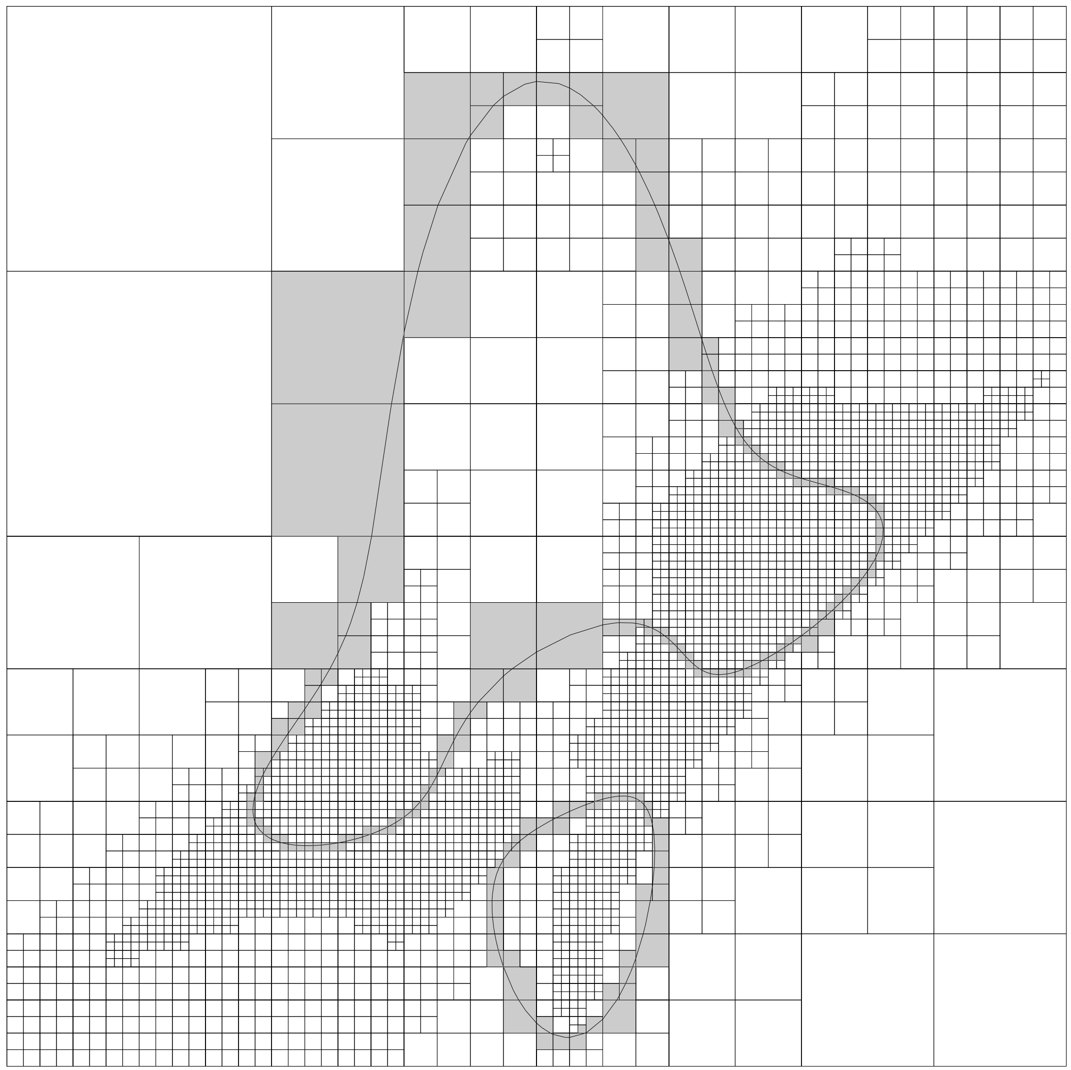
Robust adaptive approximation



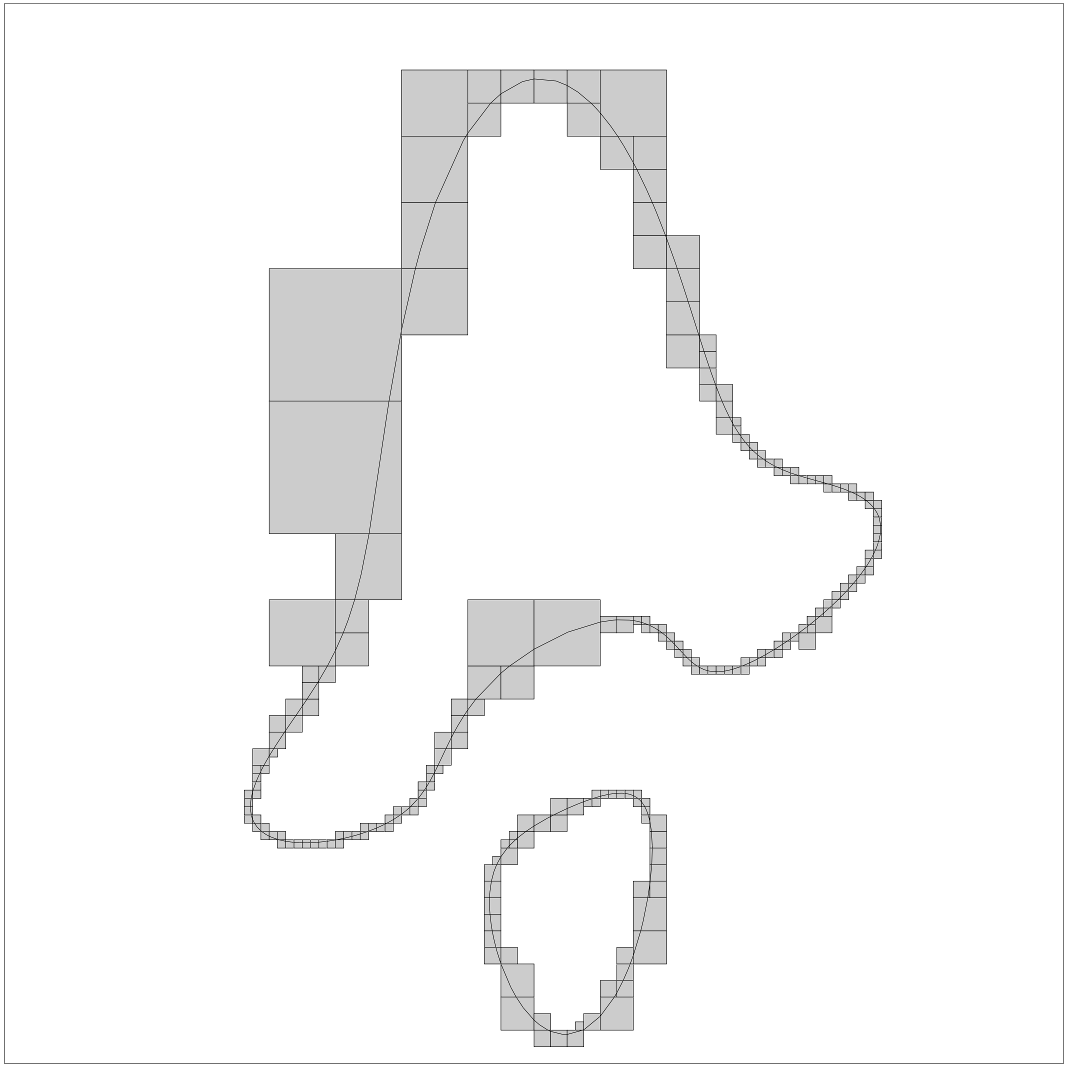
Approximation of implicit curve



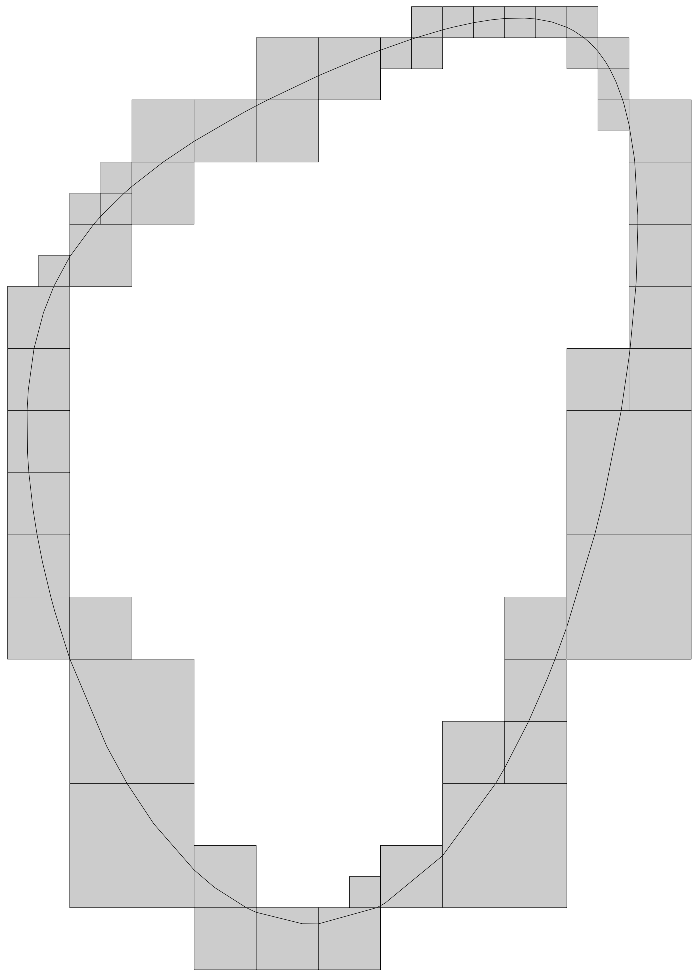
Robust adaptive approximation



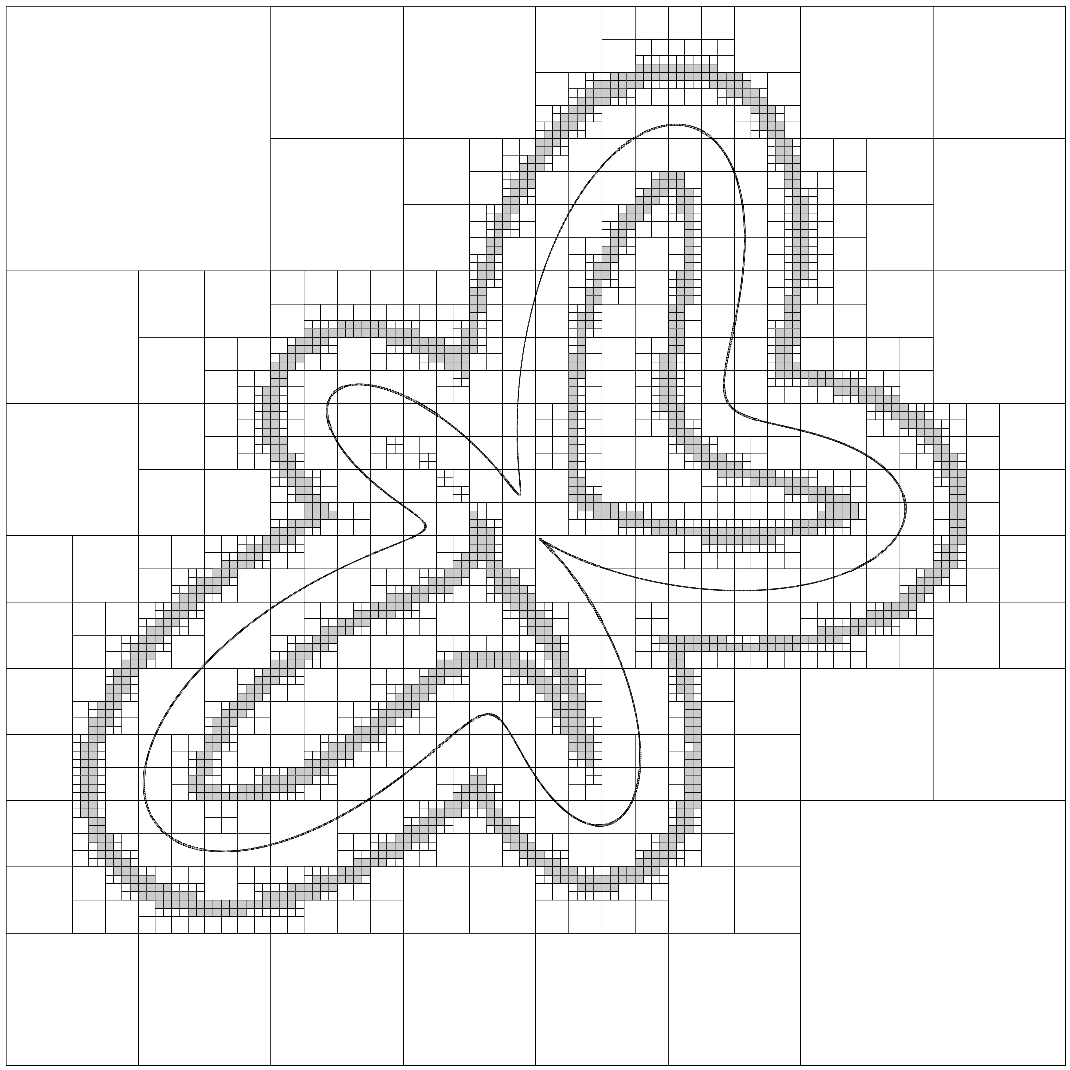
Robust adaptive approximation



Robust adaptive approximation

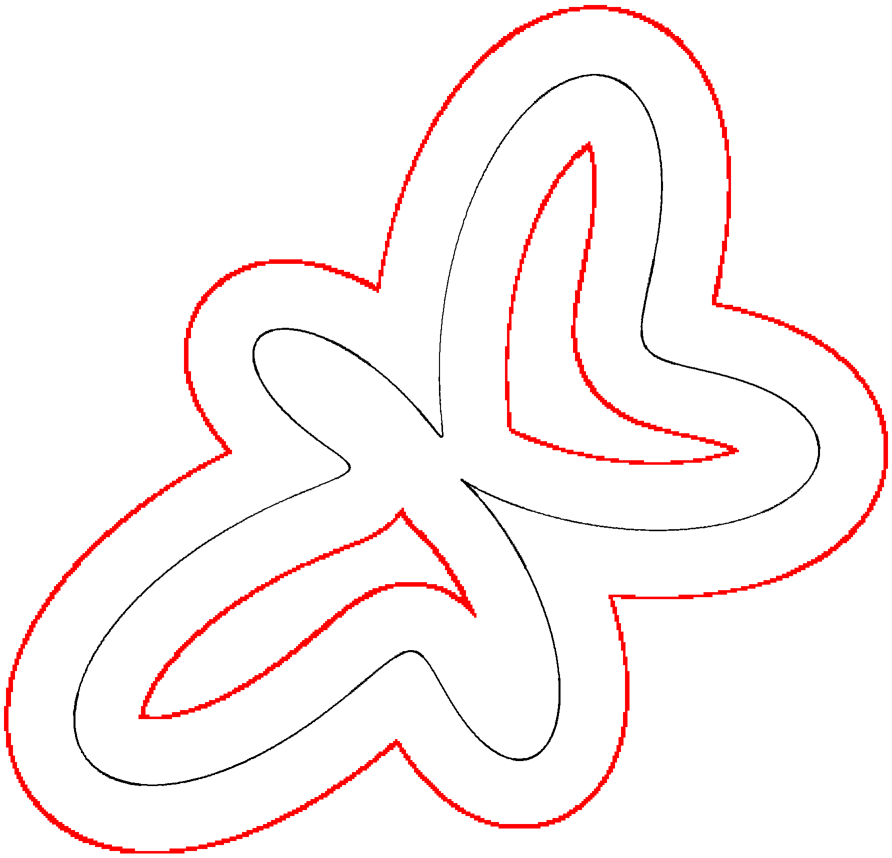


Offsets of parametric curves

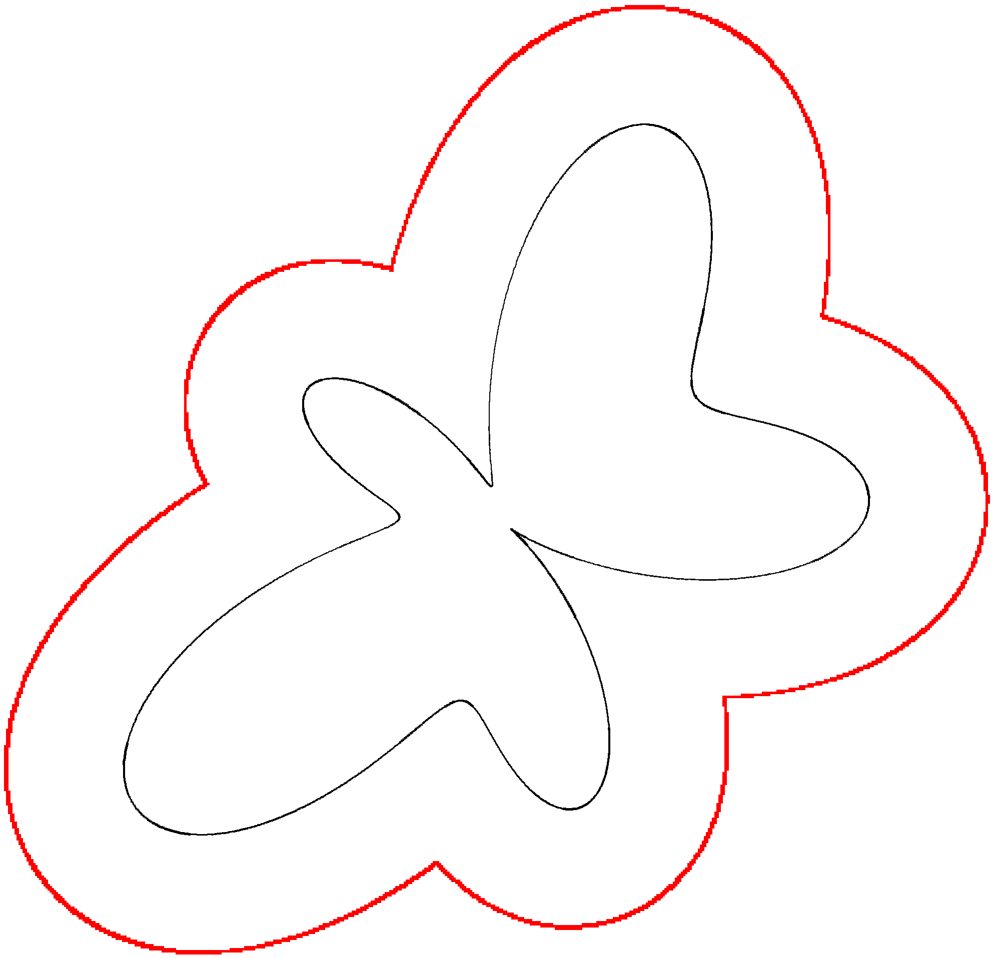


Oliveira–Figueiredo (2003)

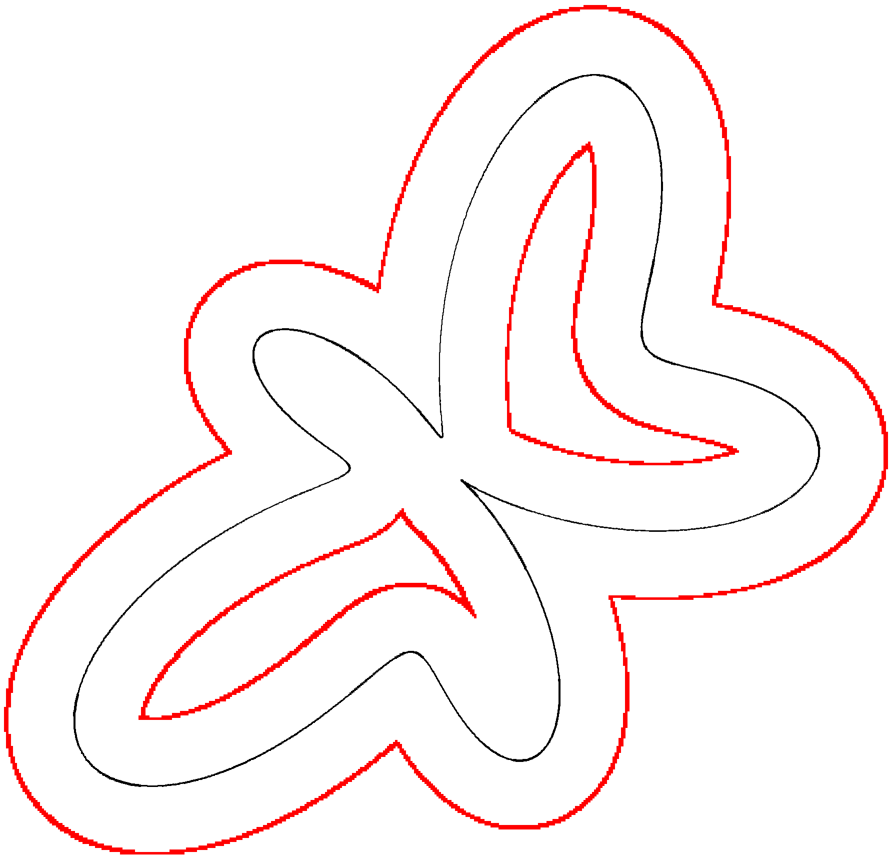
Offsets of parametric curves



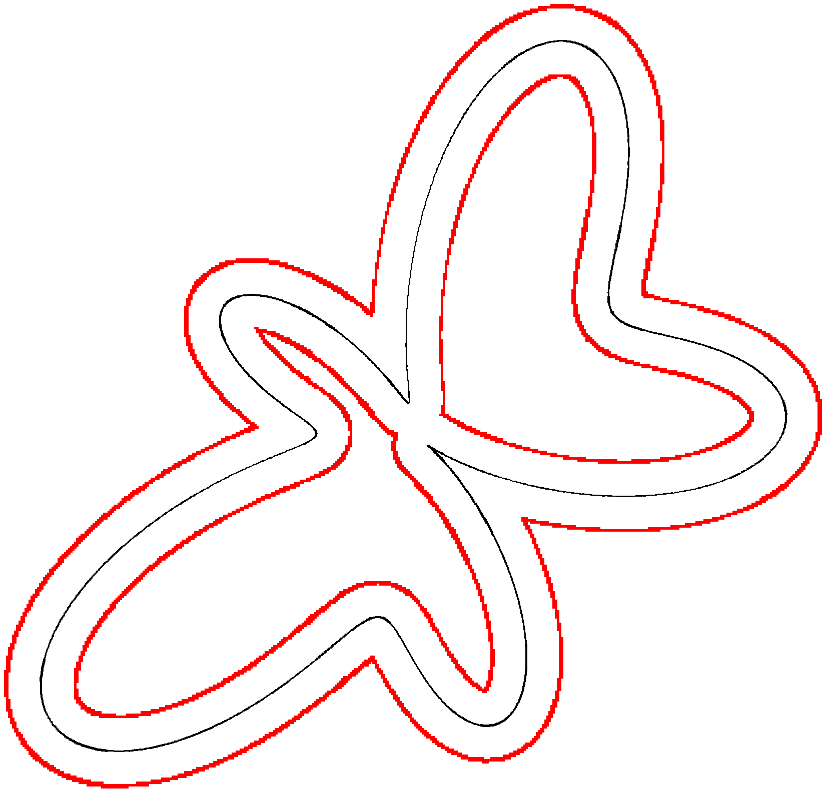
Offsets of parametric curves



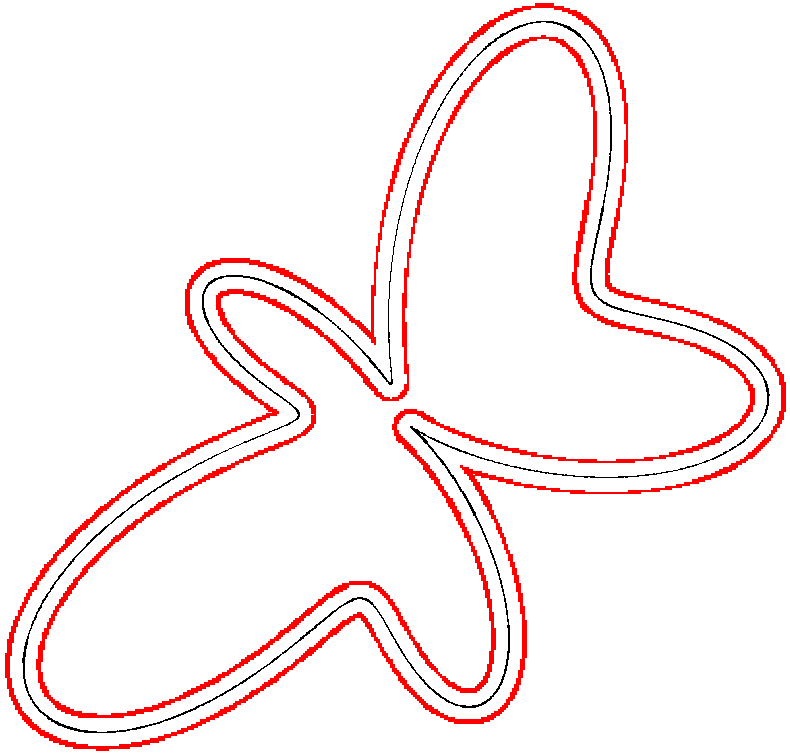
Offsets of parametric curves



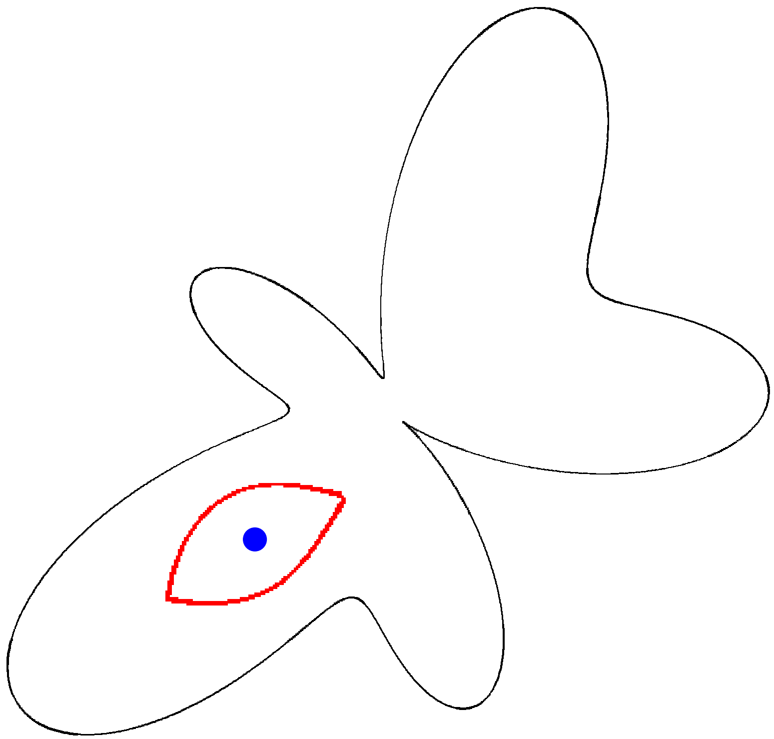
Offsets of parametric curves



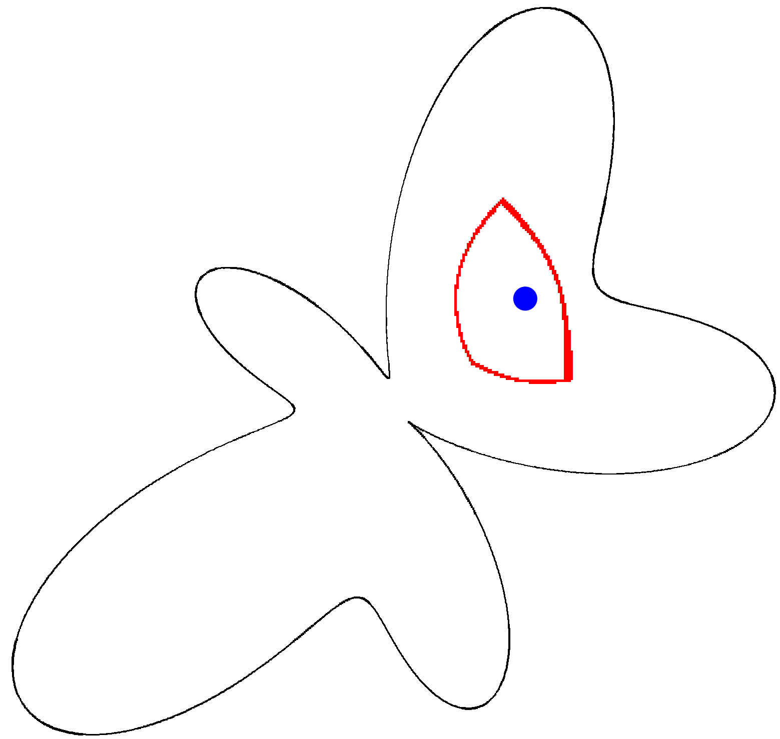
Offsets of parametric curves



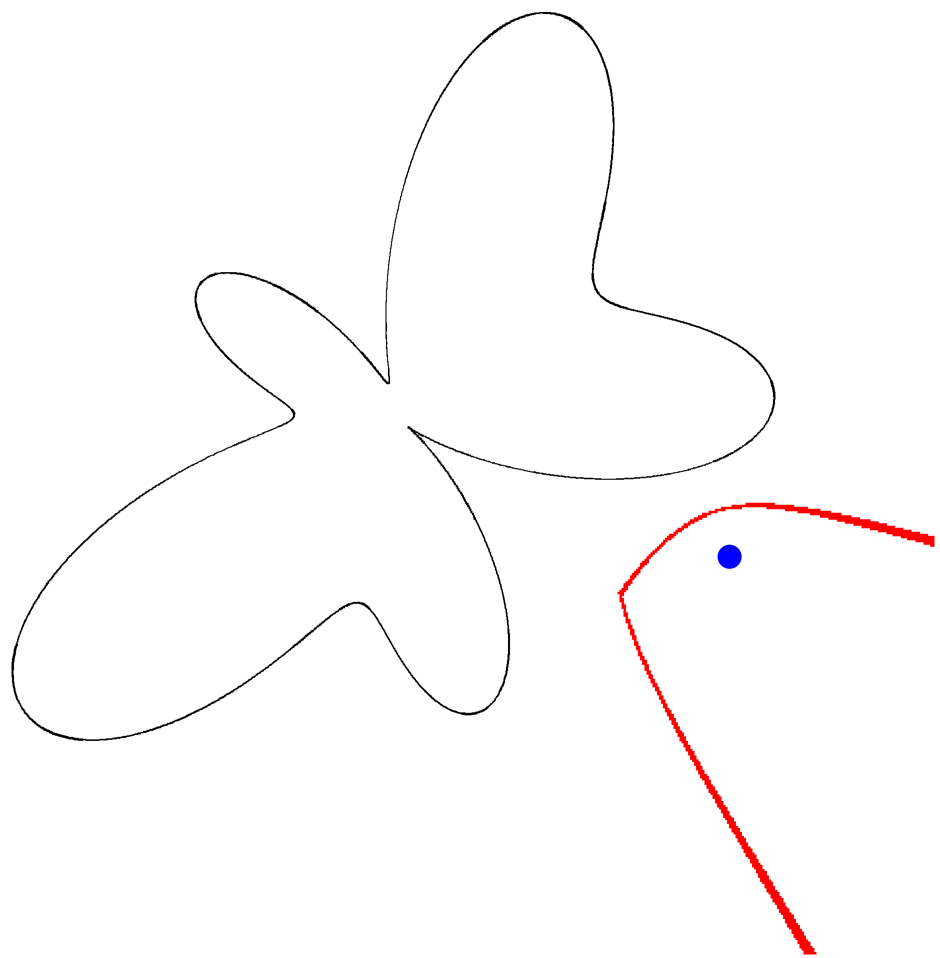
Bisectors of parametric curves



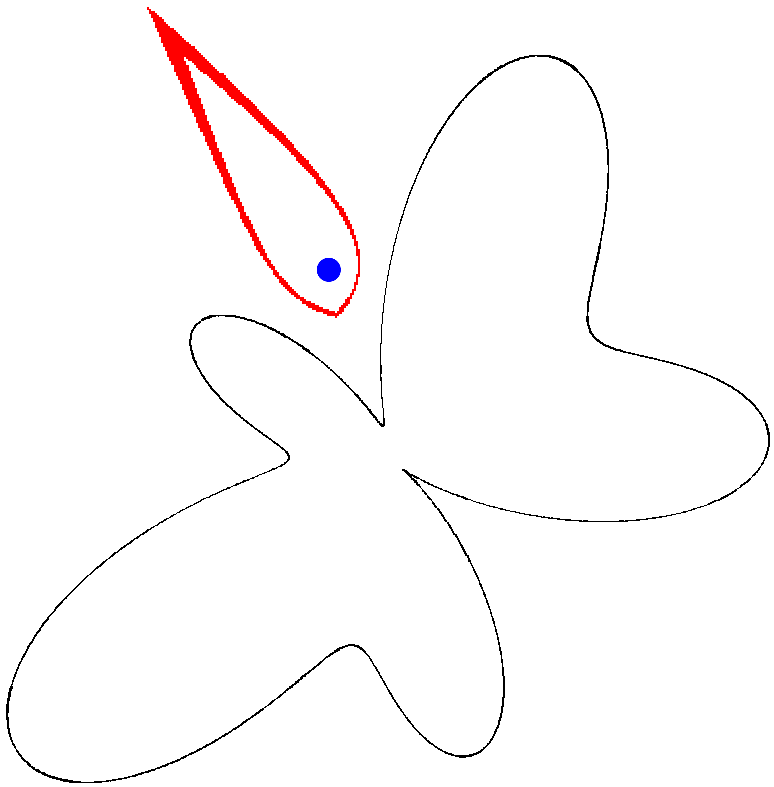
Bisectors of parametric curves



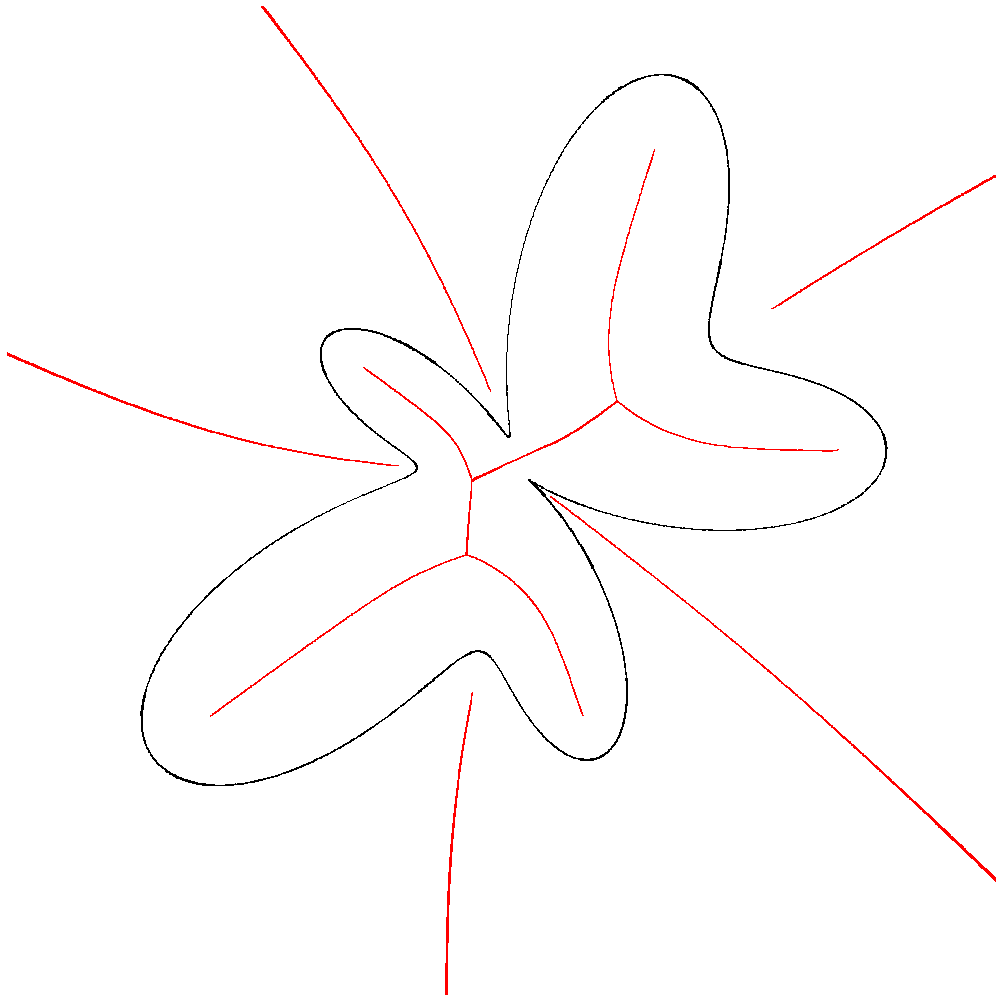
Bisectors of parametric curves



Bisectors of parametric curves



Medial axis of parametric curves



Interval methods

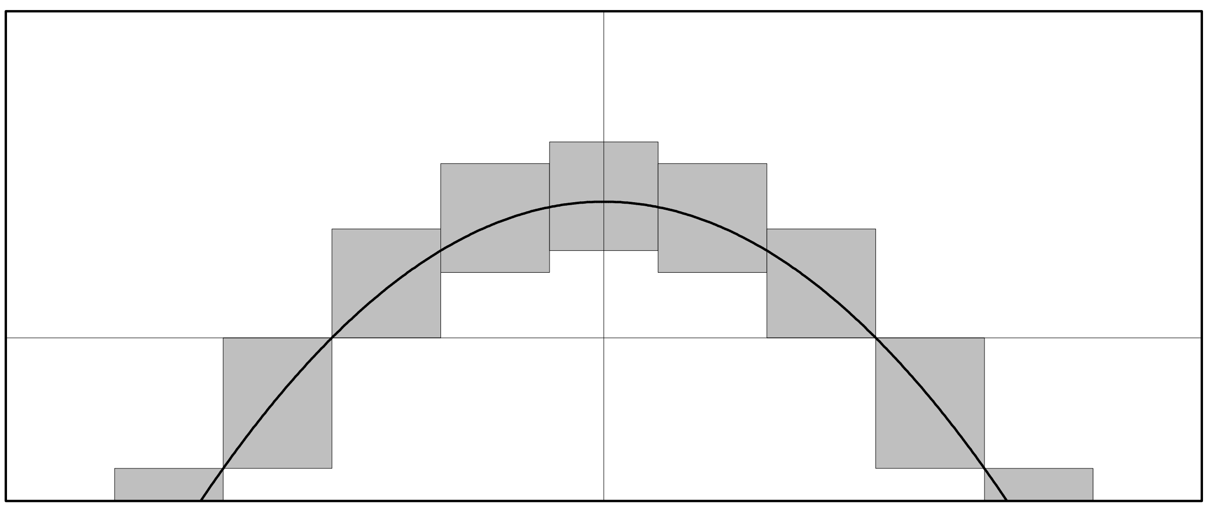
- Robust: they don't lie
  - ◇ correctness depends on $F(X) \supseteq f(X)$
  - ◇ can prove $0 \notin f(X)$, not that $0 \in f(X)$
- Converge: solutions get better
  - ◇ $F(X) \to \{f(x)\}$ as $X \to \{x\}$
- Conservative: they tend to exagerate
  - ◇ $f(x,y) = y^2 - x^3 + x \qquad X = [-2,-1] \times [1,2]$
    $F(X) = [0,11] \qquad f(X) = [1,10]$
  - ◇ gets worse in complicated expressions and iterative methods
- Efficient?
  - ◇ how much larger is $F(X)$?
  - ◇ better estimates imply faster methods

The dependency problem in interval arithmetic

IA can't see correlations between operands

$$g(x) = (10 + x)(10 - x) \text{ for } x \in [-2, 2]$$

$$\begin{aligned}
10 + x &= [8, 12] \\
10 - x &= [8, 12] \\
(10 + x)(10 - x) &= [64, 144] && \text{diam} = 80 \\
\text{Exact range} &= [96, 100] && \text{diam} = 4
\end{aligned}$$

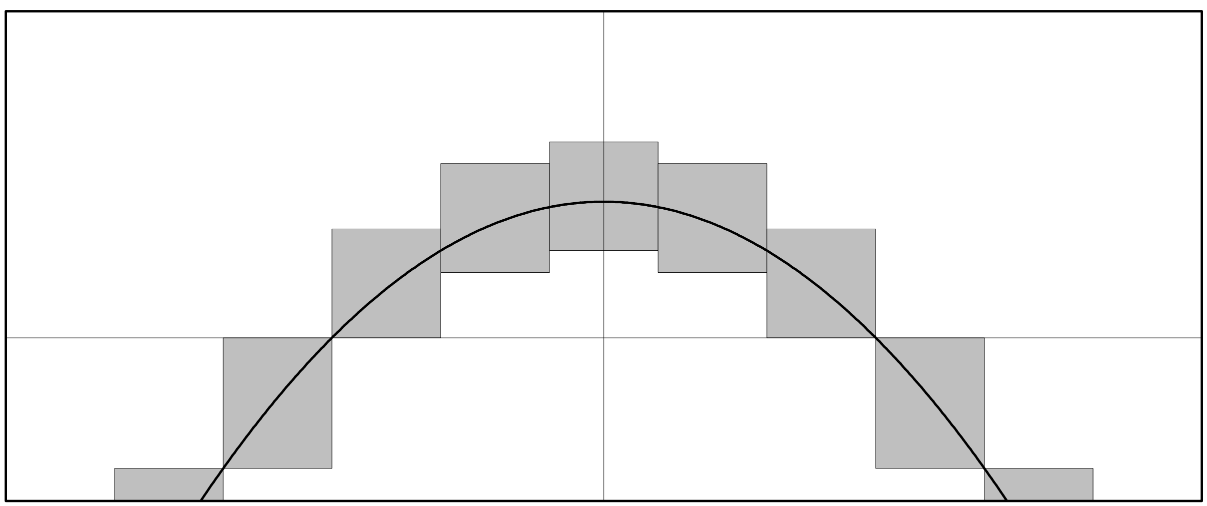


The dependency problem in interval arithmetic

IA can't see correlations between operands

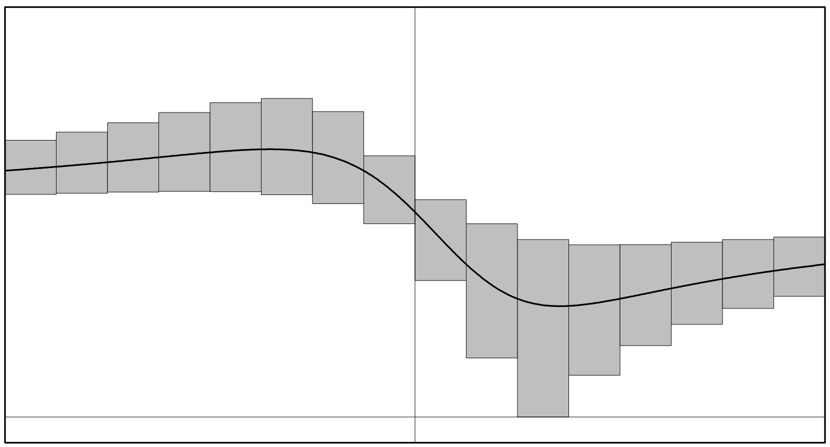
$$g(x) = (10 + x)(10 - x) \text{ for } x \in [-u, u]$$

$$\begin{aligned} 10 + x &= [10 - u, 10 + u] \\ 10 - x &= [10 - u, 10 + u] \\ (10 + x)(10 - x) &= [(10 - u)^2, (10 + u)^2] && \text{diam} = 40u \\ \text{Exact range} &= [100 - u^2, 100] && \text{diam} = u^2 \end{aligned}$$

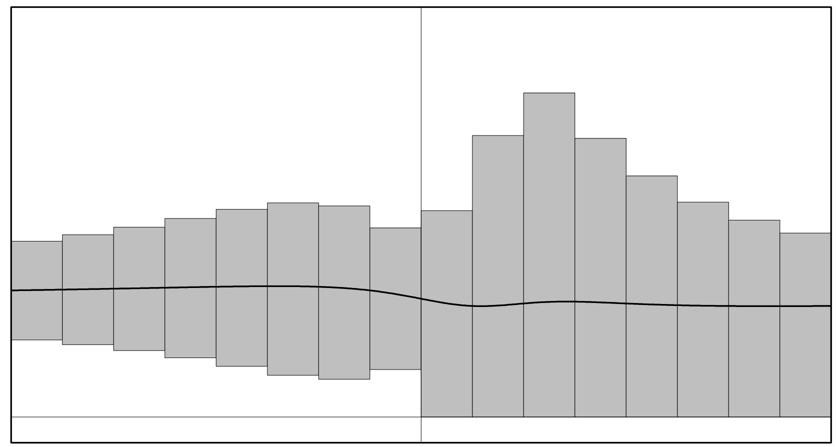


The dependency problem in interval arithmetic

$$g(x) = \sqrt{x^2 - x + 1/2}/\sqrt{x^2 + 1/2}$$



g



$g \circ g$

$g^n \to c =$ fixed point of $g \approx 0.5586$, but intervals diverge

Interval estimates may get too large in long computations

Affine arithmetic

Affine arithmetic

AA represents a quantity x with an *affine form*

$$\hat{x} = x_0 + x_1\varepsilon_1 + \cdots + x_n\varepsilon_n$$

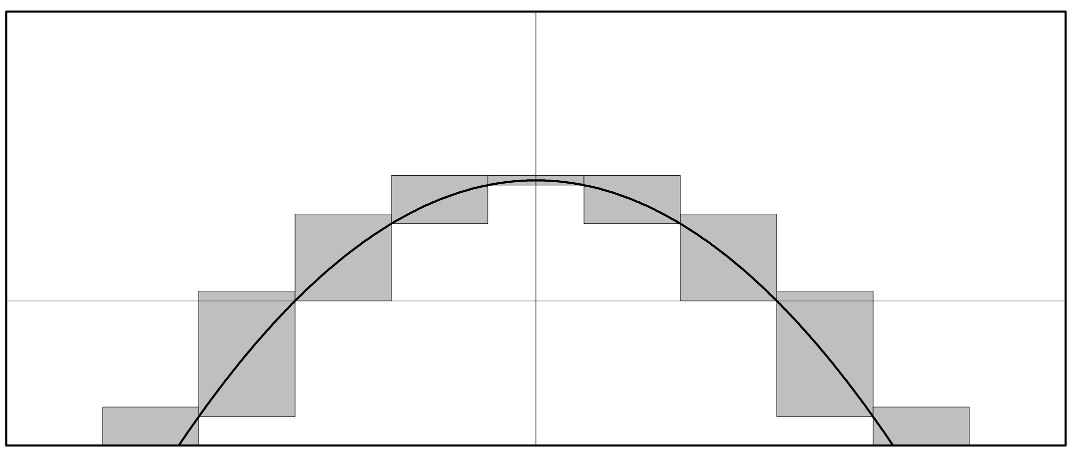
- *Noise symbols* $\varepsilon_i \in [-1, +1]$: independent, but otherwise unknown
- Can compute arbitrary formulas on affine forms
  - ◇ Need affine approximations for non-affine operations
  - ◇ New noise symbols created during computation due to approximation and rounding
- Can replace IA
  - ◇ $x \sim \hat{x} \;\Rightarrow\; x \in [x_0 - r,\, x_0 + r]$ for $r = |x_1| + \cdots + |x_n|$
  - ◇ $x \in [a, b] \;\Rightarrow\; x \sim \hat{x} = x_0 + x_1\varepsilon_1$
    $x_0 = (b + a)/2 \quad x_1 = (b - a)/2$

The dependency problem in interval arithmetic – AA version

AA can see correlations between operands

$$g(x) = (10 + x)(10 - x) \text{ for } x \in [-u, u], \qquad x = 0 + u\,\varepsilon$$

$$\begin{aligned}
10 + x &= 10 - u\,\varepsilon \\
10 - x &= 10 + u\,\varepsilon \\
(10 + x)(10 - x) &= 100 - u^2\,\varepsilon \\
\text{range} &= [100 - u^2, 100 + u^2] \qquad \text{diam} = 2u^2 \\
\text{Exact range} &= [100 - u^2, 100] \qquad \text{diam} = u^2
\end{aligned}$$



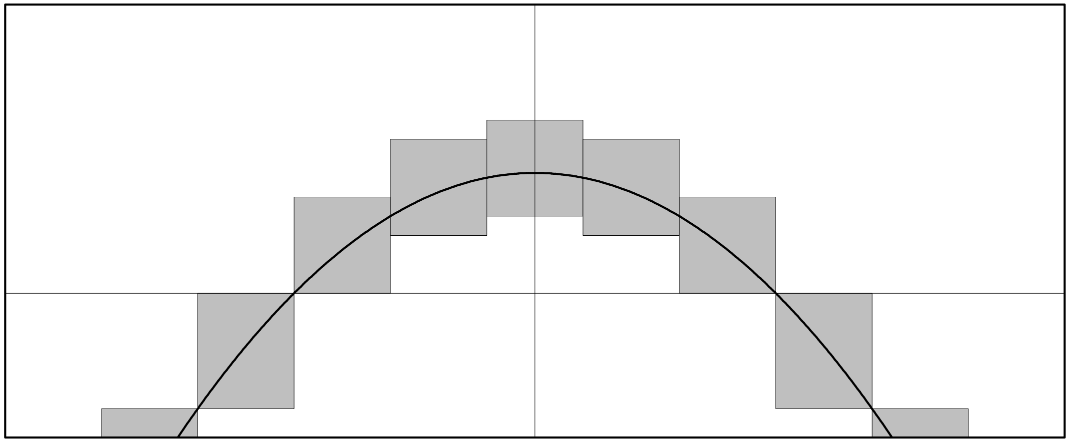
AA

The dependency problem in interval arithmetic – AA version

AA can see correlations between operands

$$g(x) = (10 + x)(10 - x) \text{ for } x \in [-u, u], \qquad x = 0 + u\,\varepsilon$$

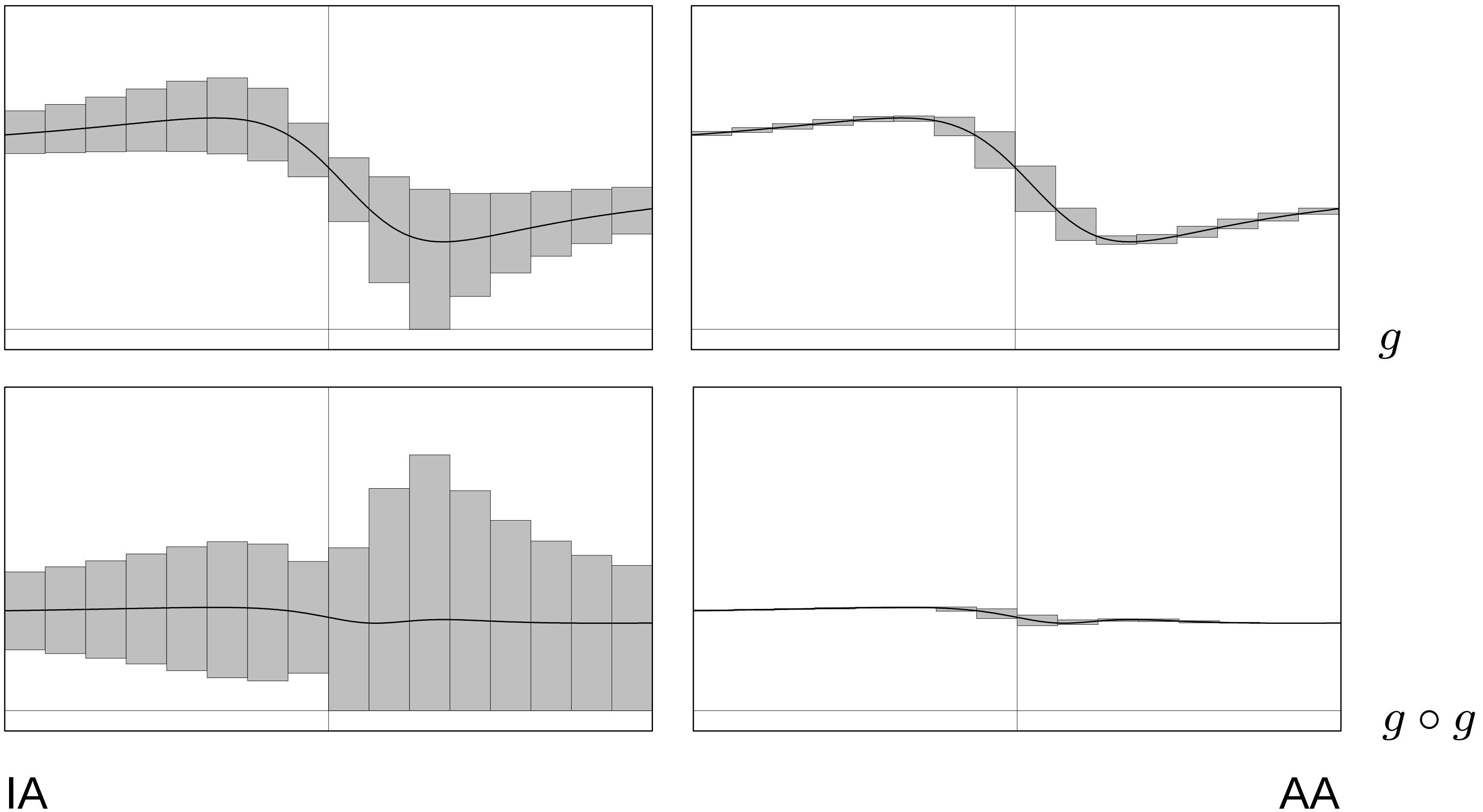
$$\begin{aligned}
10 + x &= 10 - u\,\varepsilon \\
10 - x &= 10 + u\,\varepsilon \\
(10 + x)(10 - x) &= 100 - u^2\,\varepsilon \\
\text{range} &= [100 - u^2, 100 + u^2] \qquad \text{diam} = 2u^2 \\
\text{Exact range} &= [100 - u^2, 100] \qquad \text{diam} = u^2
\end{aligned}$$



IA

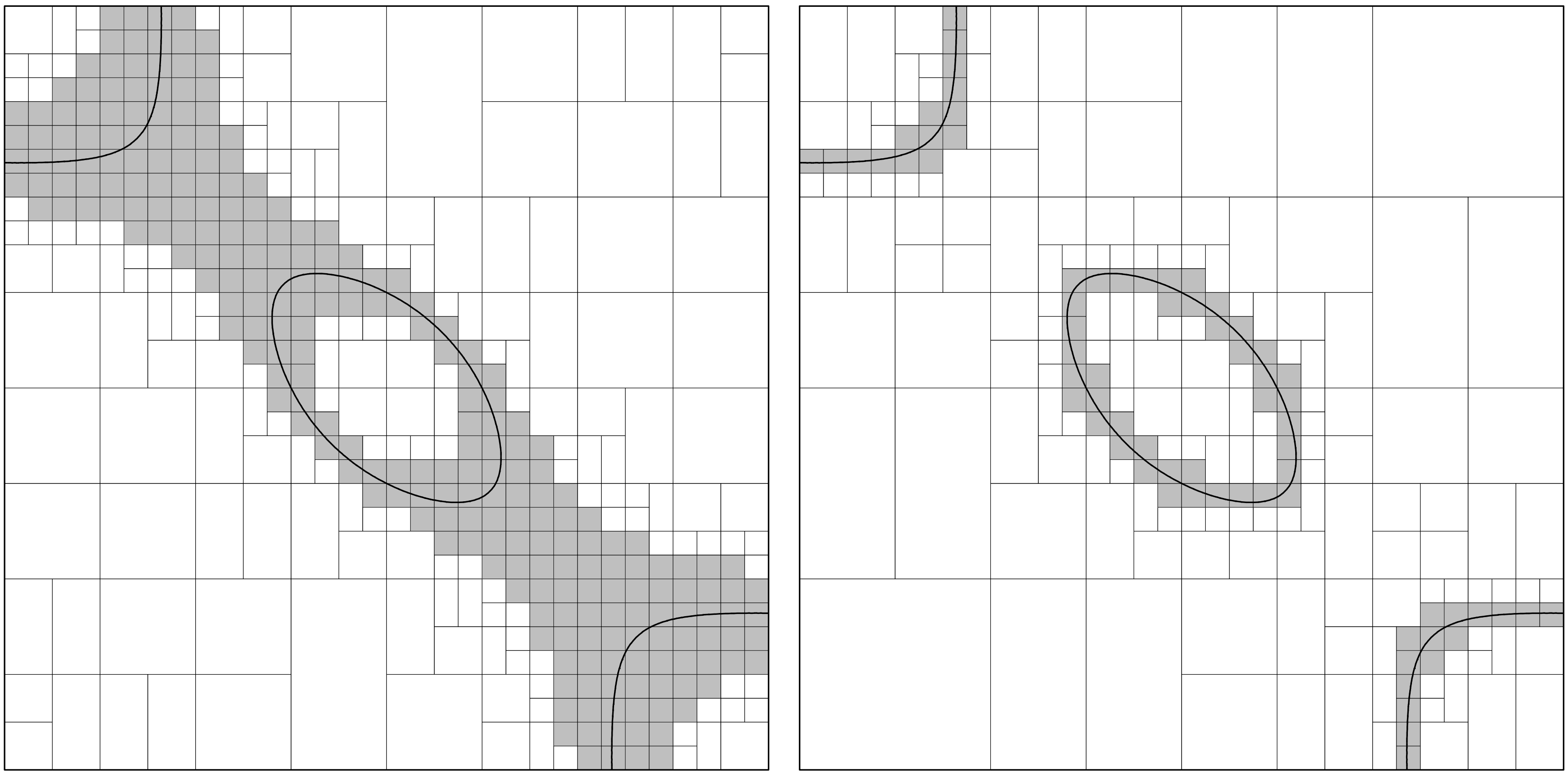
The dependency problem in interval arithmetic – AA version

$$g(x) = \sqrt{x^2 - x + 1/2}/\sqrt{x^2 + 1/2}$$



Replacing IA with AA for plotting implicit curves

$$x^2 + y^2 + xy - (xy)^2/2 - 1/4 = 0$$

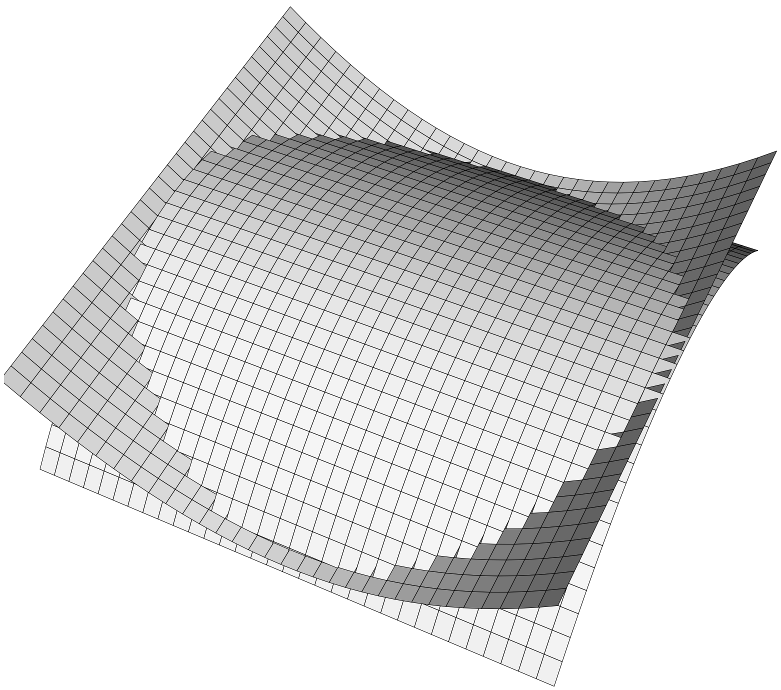


IA (246 cells, 66 exact) (Comba–Stolfi , 1993) (70 cells) AA

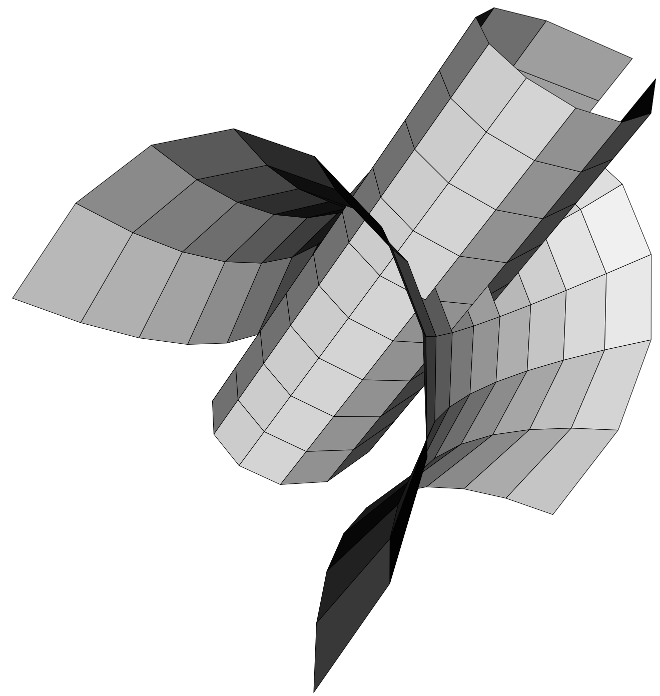
Replacing IA with AA for surface intersection

Tensor product Bézier surfaces of degree (p, q):

$$f(u,v) = \sum_{i=0}^{p} \sum_{j=0}^{q} a_{ij} B_i^p(u) B_j^q(v), \quad B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i}$$



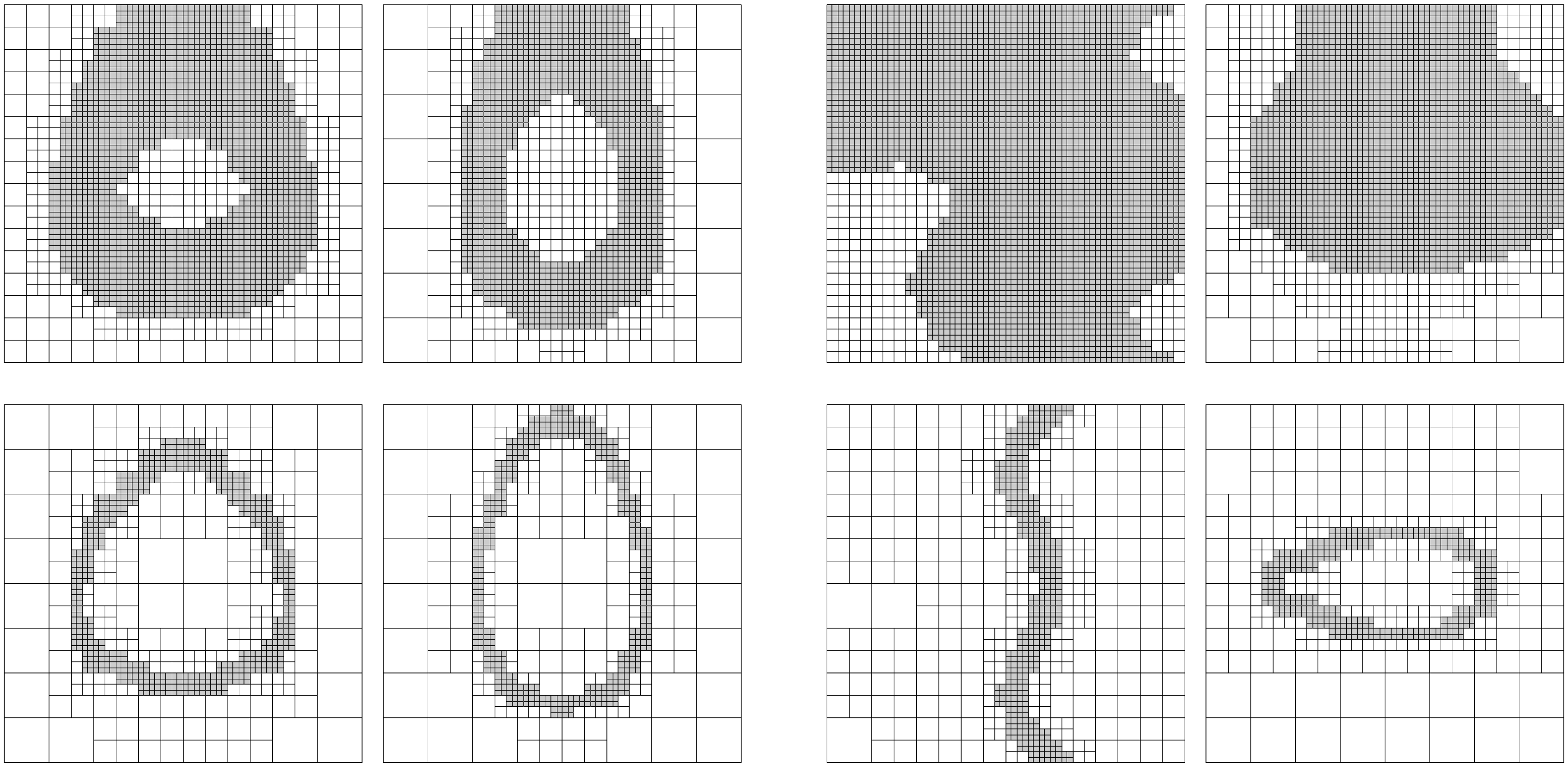
$(2, 1)$



$(3, 3)$

Replacing IA with AA for surface intersection

$(2,1)$ IA $(3,3)$



AA

(Figueiredo, 1996)

Exploiting the correlations given by AA

Geometry of affine arithmetic

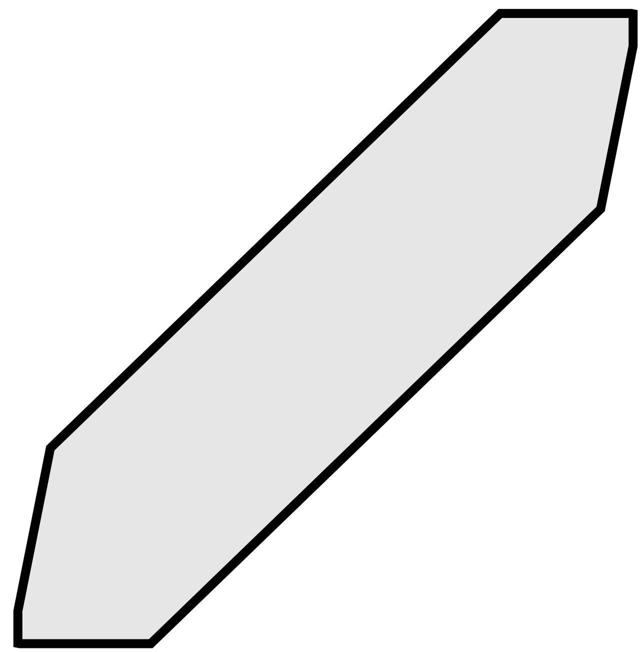
Affine forms that share noise symbols are not independent:

$$\hat{x} = x_0 + x_1\varepsilon_1 + \cdots + x_n\varepsilon_n$$
$$\hat{y} = y_0 + y_1\varepsilon_1 + \cdots + y_n\varepsilon_n$$

The region containing (x, y) is

$$Z = \{(x, y) : \varepsilon_i \in \mathbf{U}\}$$

This region is the image of $\mathbf{U}^n$ under an affine map $\mathbf{R}^n \to \mathbf{R}^2$. It's a centrally symmetric convex polygon, a *zonotope*.



Geometry of affine arithmetic

Affine forms that share noise symbols are not independent:

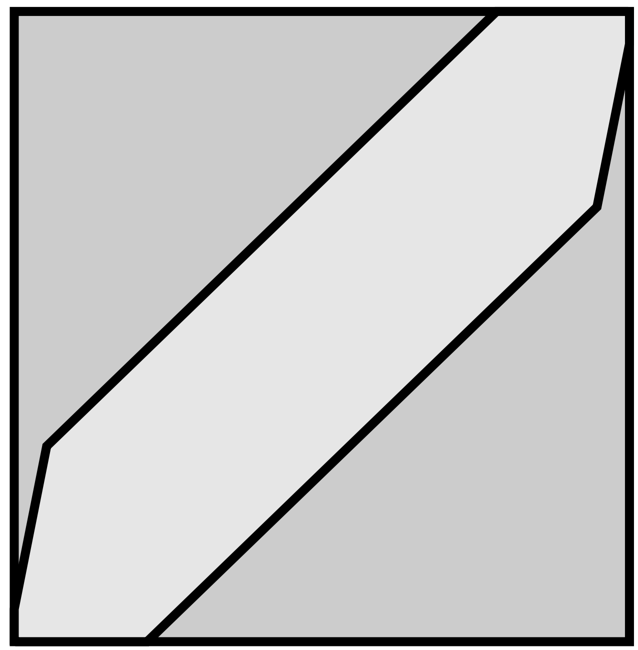
$$\hat{x} = x_0 + x_1\varepsilon_1 + \cdots + x_n\varepsilon_n$$
$$\hat{y} = y_0 + y_1\varepsilon_1 + \cdots + y_n\varepsilon_n$$

The region containing (x, y) is

$$Z = \{(x, y) : \varepsilon_i \in \mathbf{U}\}$$

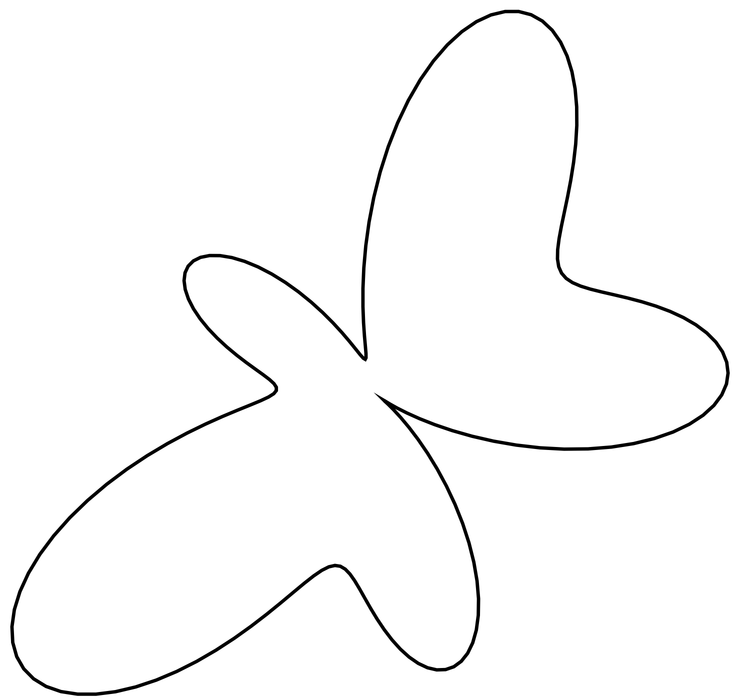
This region is the image of $\mathbf{U}^n$ under an affine map $\mathbf{R}^n \to \mathbf{R}^2$. It's a centrally symmetric convex polygon, a *zonotope*.

The region would be a rectangle if x and y were independent.



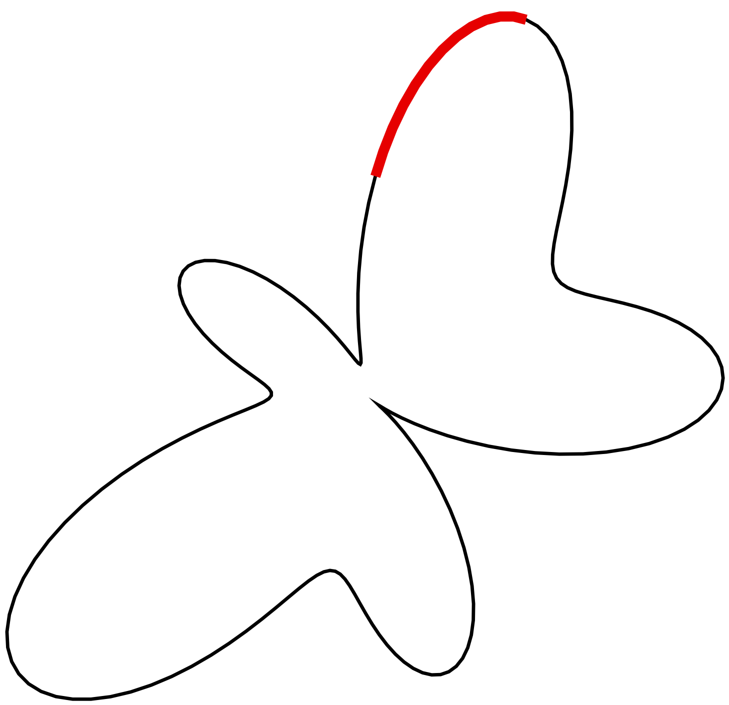
Approximating parametric curves

Given a parametric curve $\mathcal{C} = \gamma(I)$, where $\gamma\colon I \to \mathbf{R}^2$ and $T \subseteq I$, compute a bounding rectangle for $\mathcal{P} = \gamma(T)$.



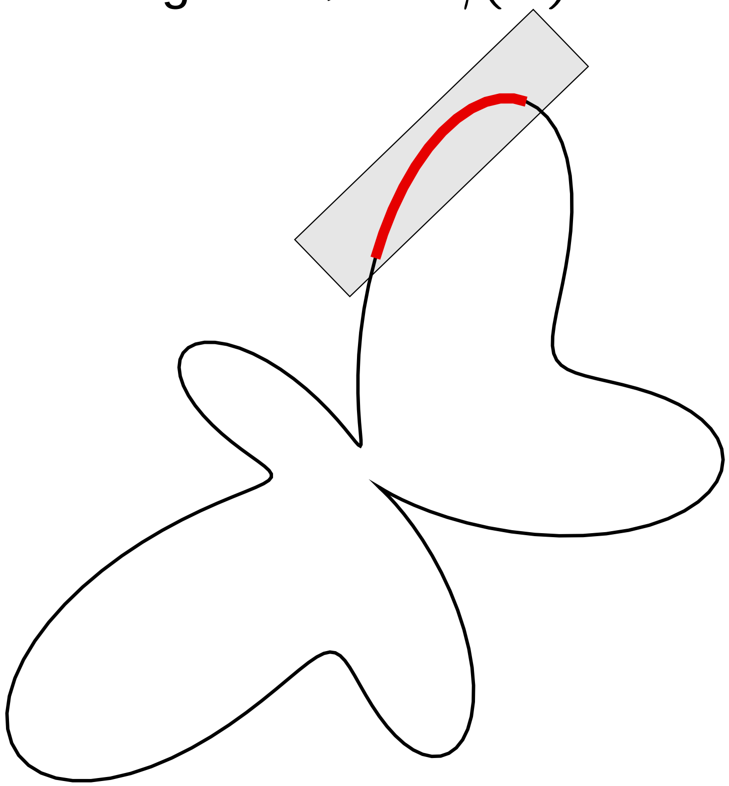
Approximating parametric curves

Given a parametric curve $\mathcal{C} = \gamma(I)$, where $\gamma\colon I \to \mathbf{R}^2$ and $T \subseteq I$, compute a bounding rectangle for $\mathcal{P} = \gamma(T)$.



Approximating parametric curves

Given a parametric curve $\mathcal{C} = \gamma(I)$, where $\gamma\colon I \to \mathbf{R}^2$ and $T \subseteq I$, compute a bounding rectangle for $\mathcal{P} = \gamma(T)$.



Approximating parametric curves

Given a parametric curve $\mathcal{C} = \gamma(I)$, where $\gamma: I \to \mathbf{R}^2$ and $T \subseteq I$, compute a bounding rectangle for $\mathcal{P} = \gamma(T)$.

Solution:

- Write $\gamma(t) = (x(t), y(t))$.
- Represent $t \in T$ with an affine form:

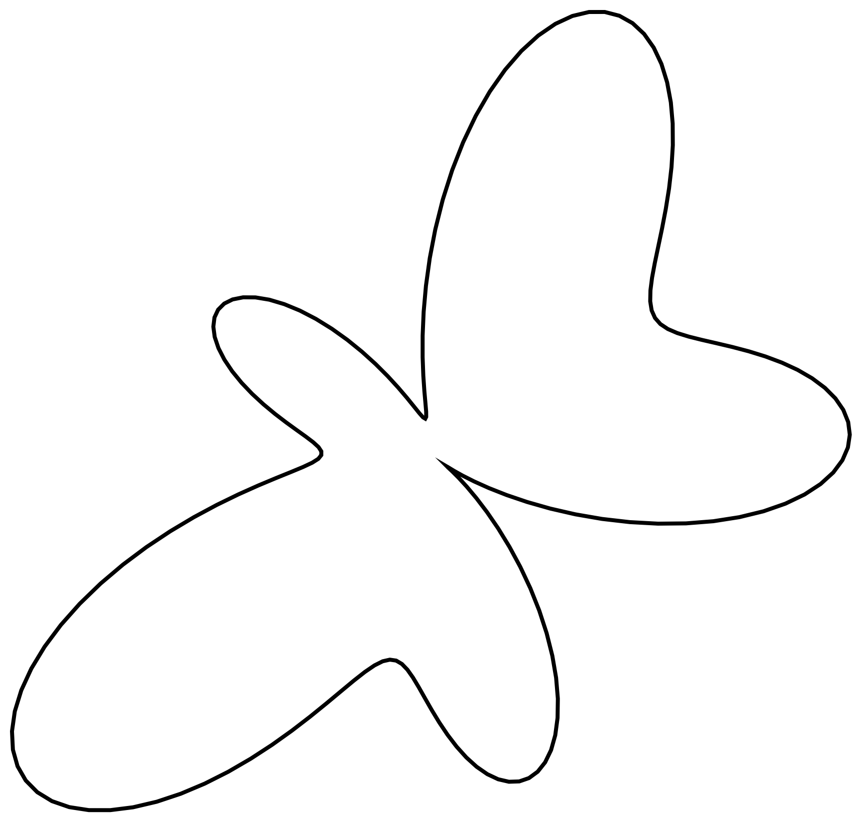
$$\hat{t} = t_0 + t_1\,\varepsilon_1, \qquad t_0 = (b+a)/2, \quad t_1 = (b-a)/2$$

- Compute coordinate functions x and y at $\hat{t}$ using AA:

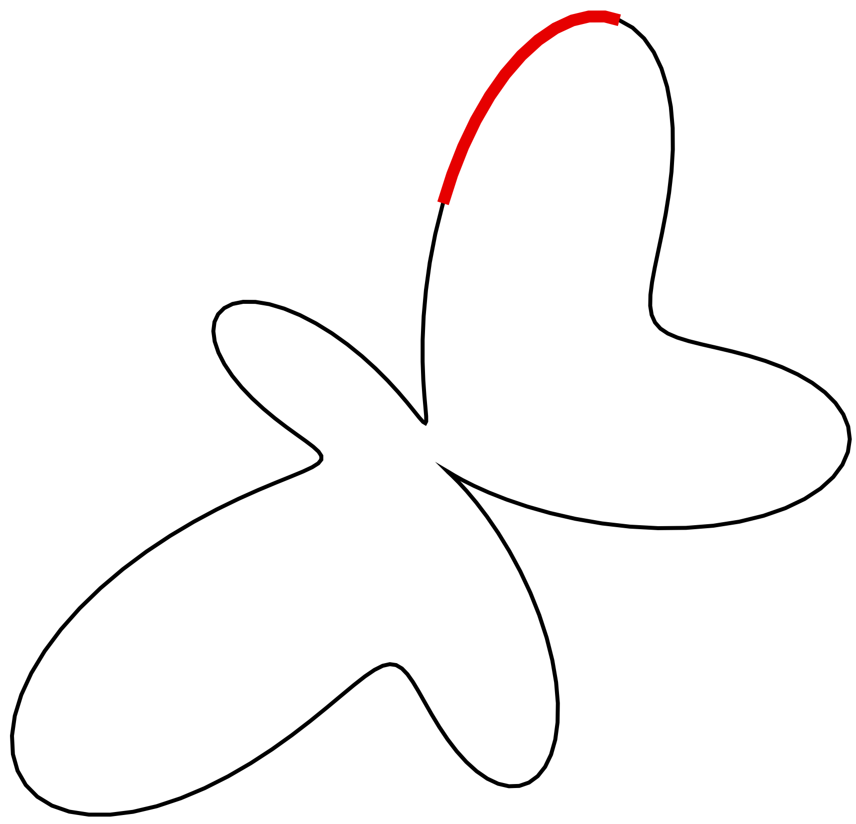
$$\begin{aligned} \hat{x} &= x_0 + x_1\varepsilon_1 + \cdots + x_n\varepsilon_n \\ \hat{y} &= y_0 + y_1\varepsilon_1 + \cdots + y_n\varepsilon_n \end{aligned}$$

- Use bounding rectangle of the xy zonotope.

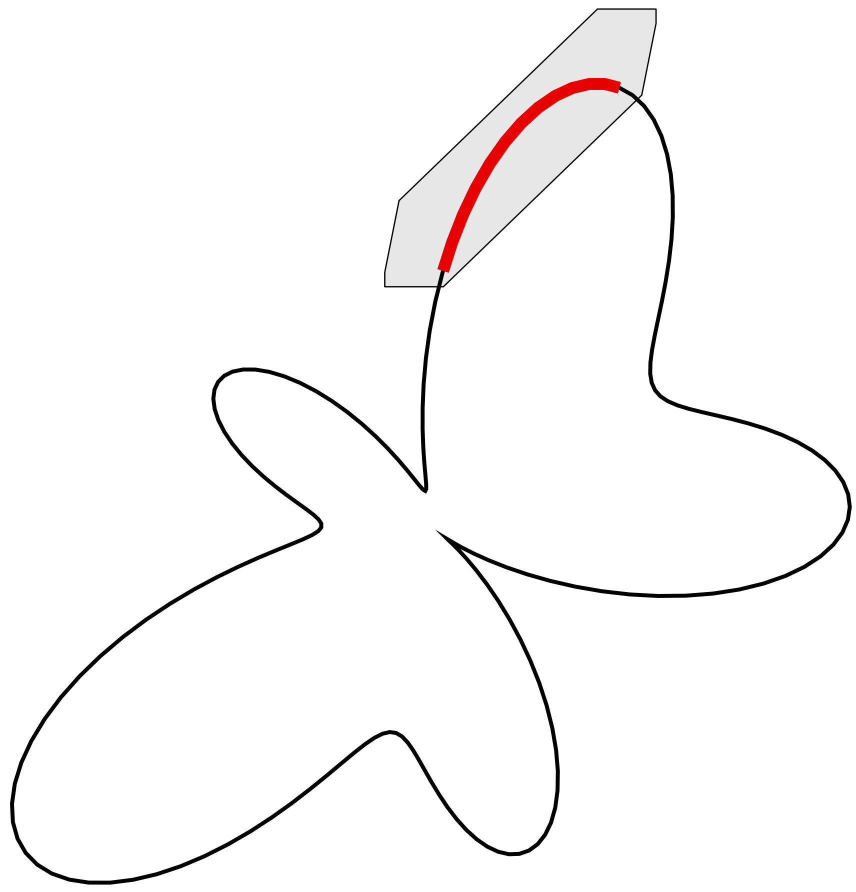
Approximating parametric curves



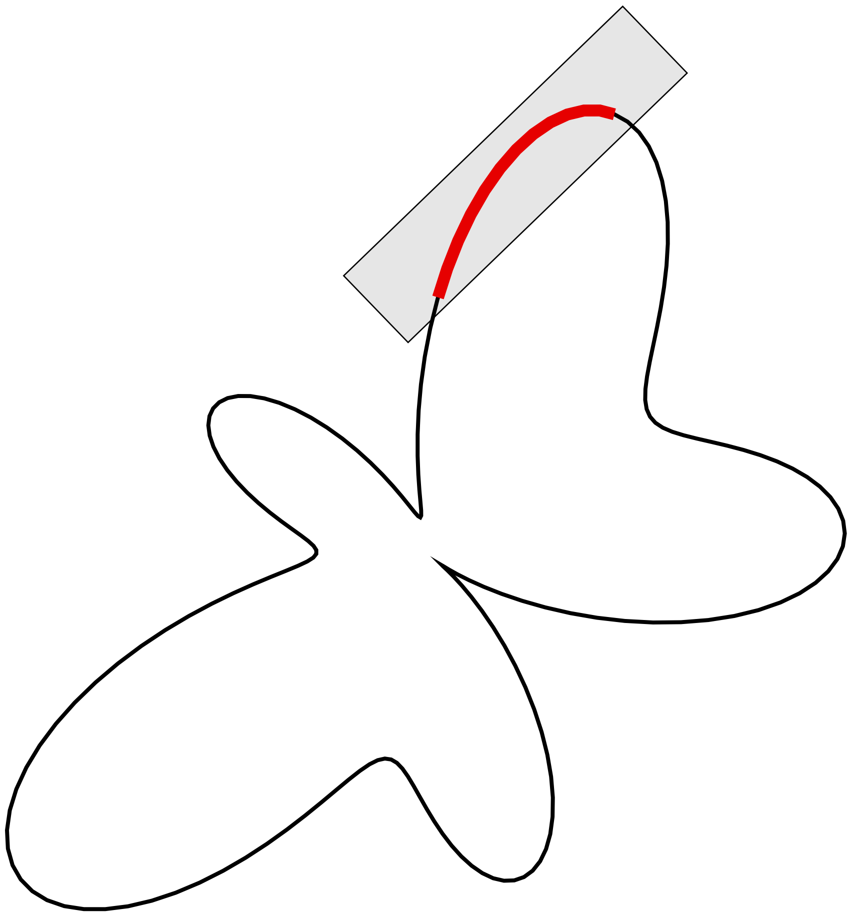
Approximating parametric curves



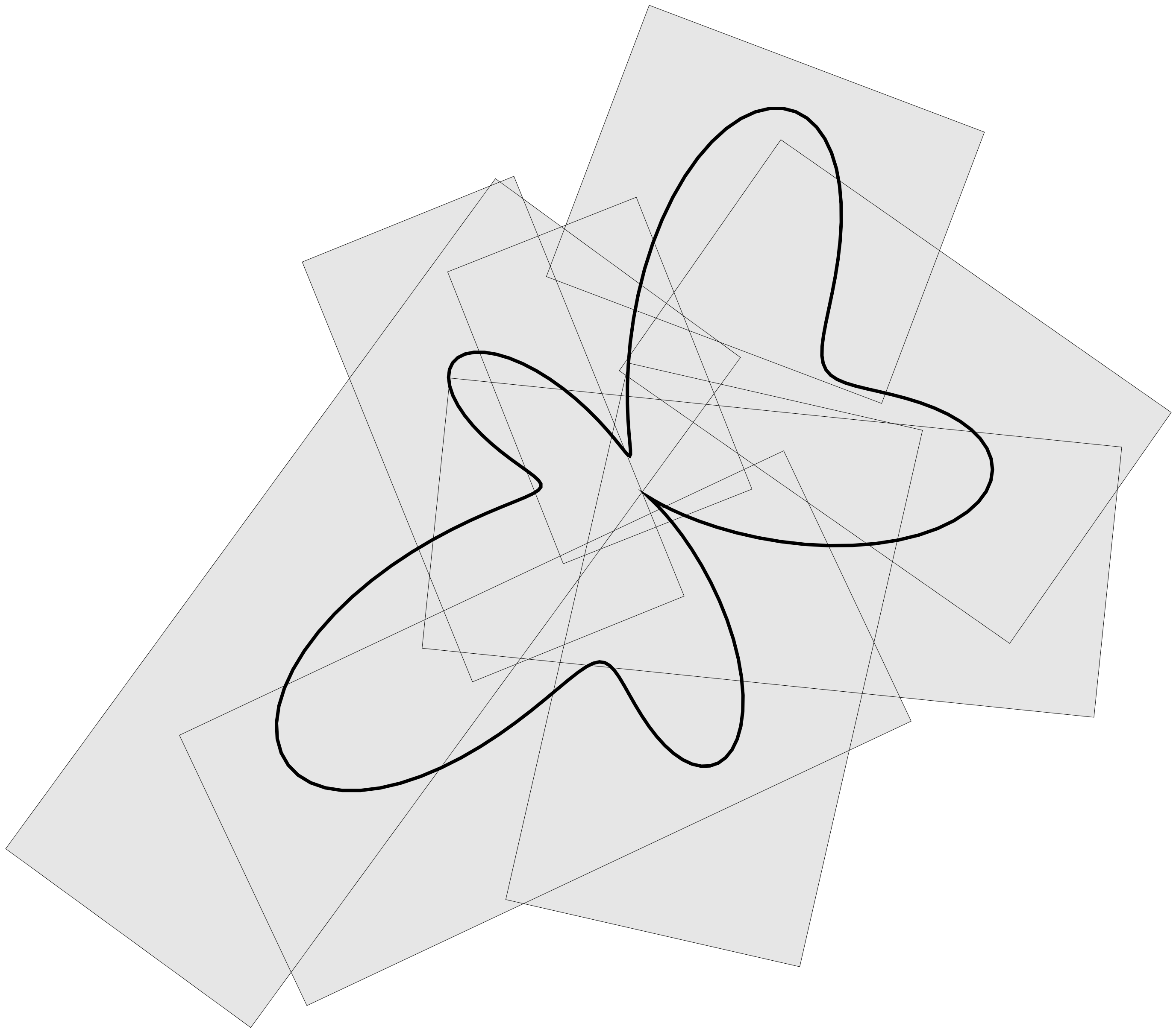
Approximating parametric curves



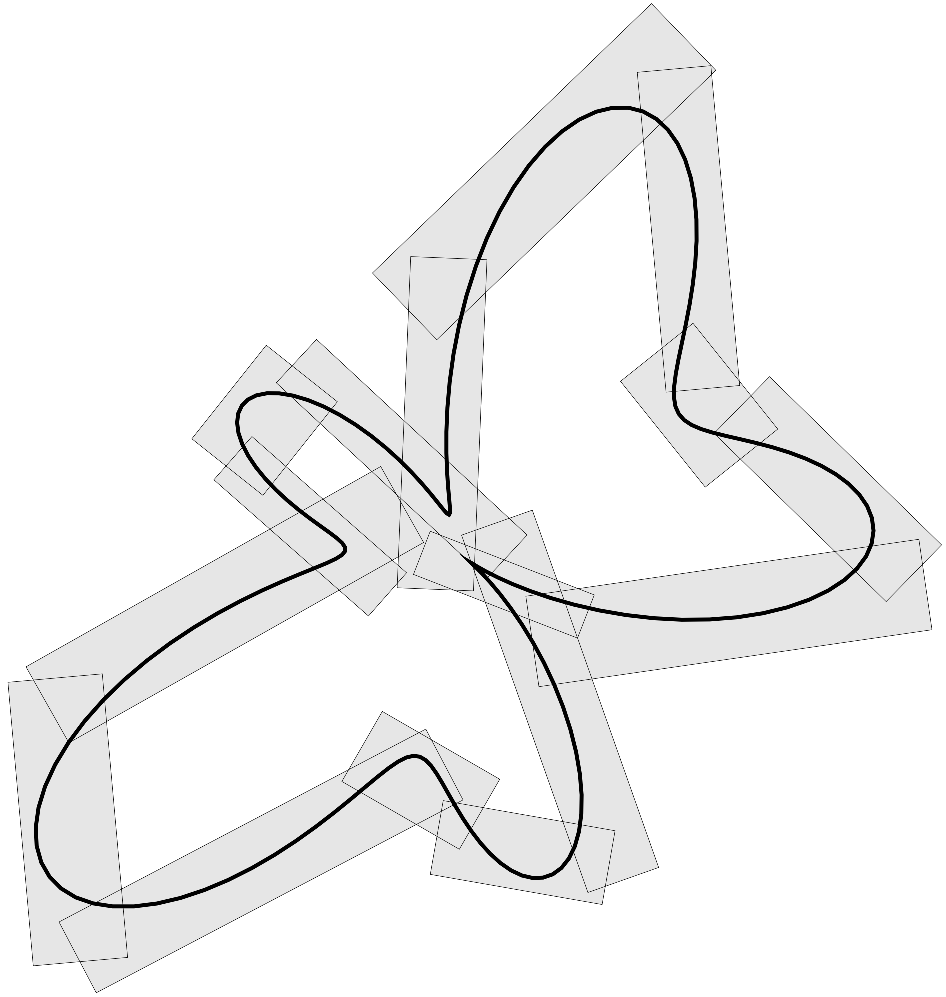
Approximating parametric curves



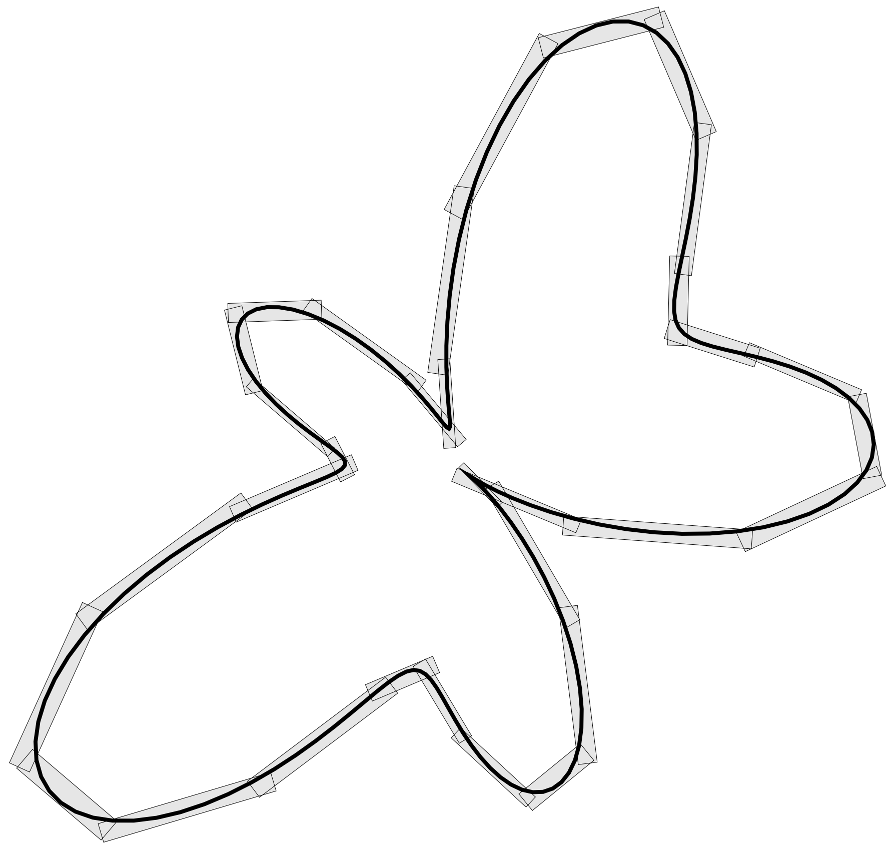
Approximating parametric curves



Approximating parametric curves



Approximating parametric curves

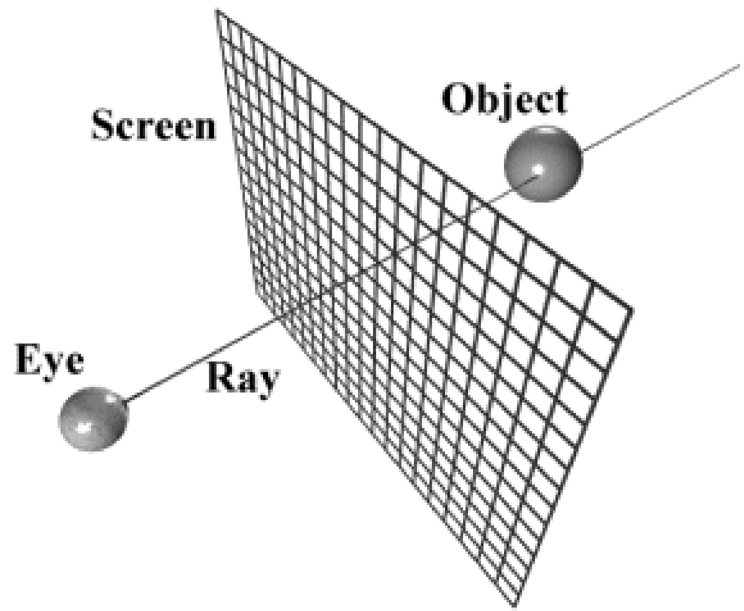


Approximating parametric curves

(Figueiredo–Stolfi –Velho, 2003)

Ray casting implicit surfaces

- Implicit surface
  $h: \mathbf{R}^3 \to \mathbf{R}$
  $S = \{p \in \mathbf{R}^3 : h(p) = 0\}$
- Ray
  $r(t) = E + t \cdot v, \quad t \in [0, \infty)$
- Ray intersects S when
  $f(t) = h(r(t)) = 0$
- First intersection occurs at *smallest* zero of f in $[0, \infty)$.
- Paint pixel with color based on normal at first intersection point



Ray casting implicit surfaces

- Implicit surface

  $h\colon \mathbf{R}^3 \to \mathbf{R}$

  $S = \{p \in \mathbf{R}^3 : h(p) = 0\}$

- Ray

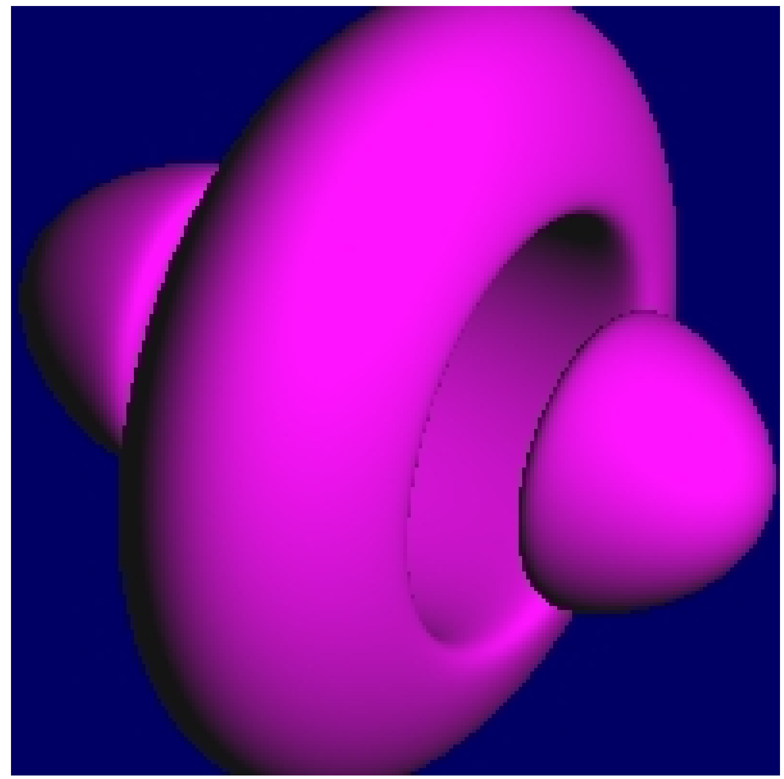
  $r(t) = E + t \cdot v, \quad t \in [0, \infty)$

- Ray intersects S when

  $f(t) = h(r(t)) = 0$

- First intersection occurs at *smallest* zero of f in $[0, \infty)$.

- Paint pixel with color based on normal at first intersection point



$4(x^4 + (y^2 + z^2)^2) + 17x^2(y^2 + z^2) - 20(x^2 + y^2 + z^2) + 17 = 0$

(Custatis–Figueiredo–Gattass, 1999)

Interval bisection

- Solve $f(t) = 0$ using inclusion function F for f:

$$F(T) \supseteq f(T) = \{f(t) : t \in T\}, \quad T \subseteq I$$

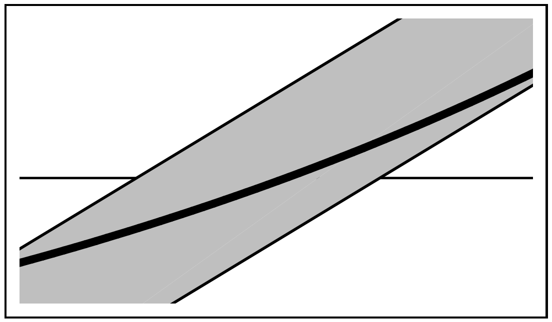
- $0 \notin F(T) \Rightarrow$ *no* solutions of $f(t) = 0$ in T
- $0 \in F(T) \Rightarrow$ there *may* be solutions in T

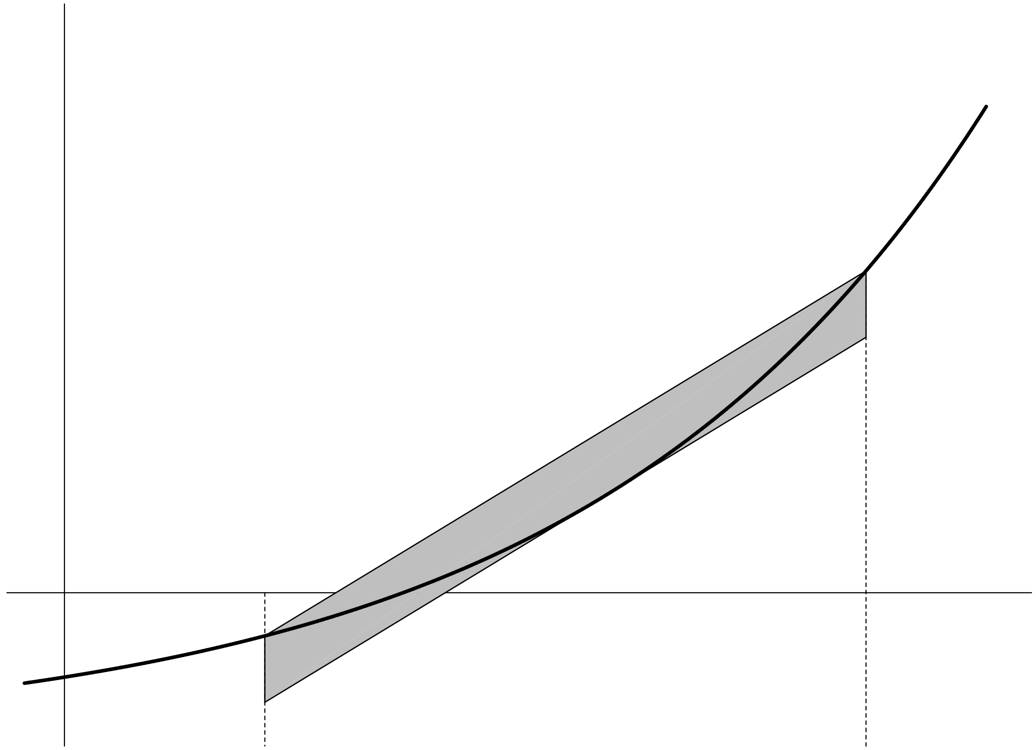
```
interval-bisection([a, b]):
    if 0 ∈ F([a, b]) then
        c ← (a + b)/2
        if (b − a) < ε then
            return c
        else
            interval-bisection([a, c])        ← try left half first!
            interval-bisection([c, b])
```

Start with interval-bisection($[0, t_\infty]$) to find the *first* zero.

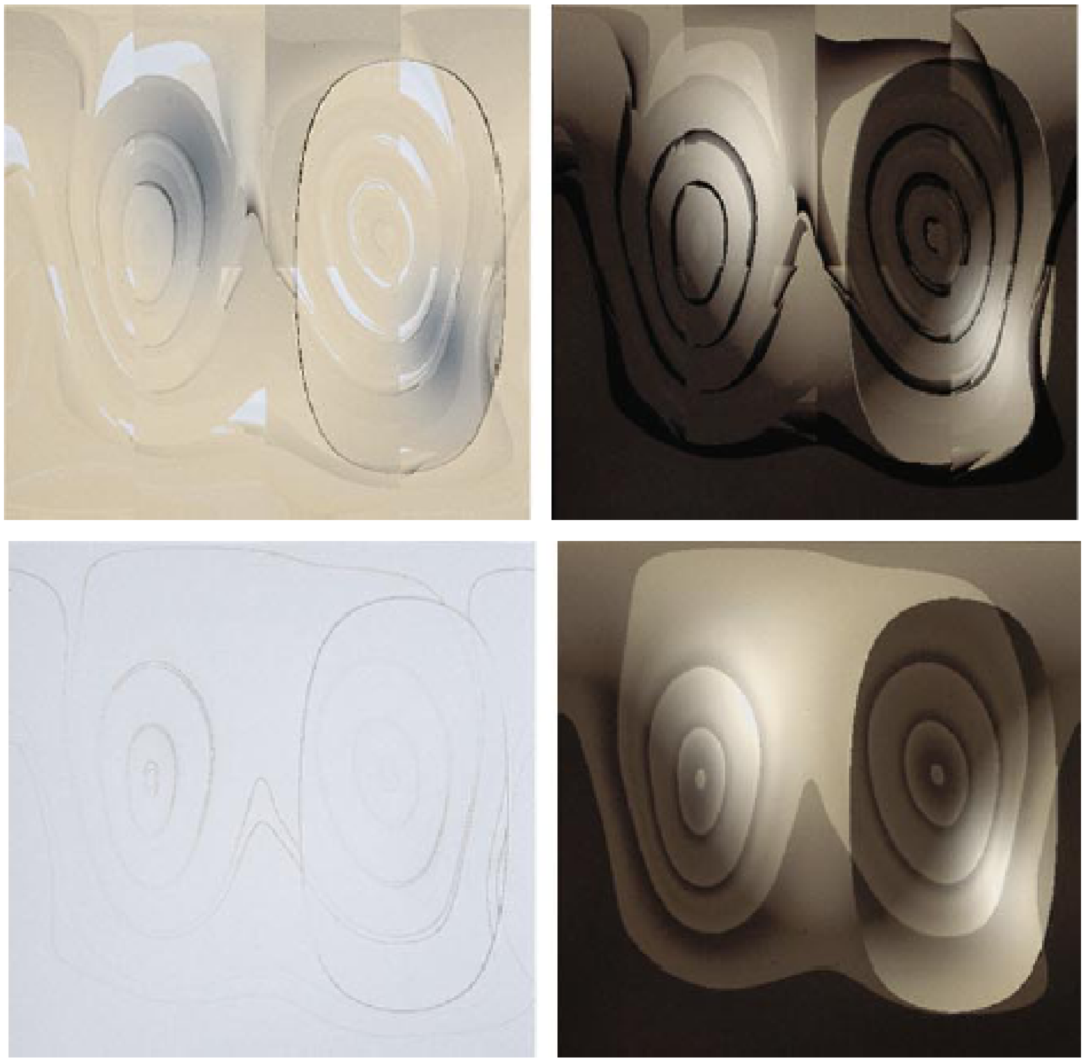
Ray casting implicit surfaces with affine arithmetic

- AA exploits linear correlations of x, y, z in $f(t) = h(r(t))$
- AA provides additional information
  - ◇ root must lie in smaller interval
  - ◇ quadratic convergence near simple zeros





Sampling procedural shaders



IA

AA (Heidrich–Slusallek–Seidel)

Conclusion

Interval methods have a place for solving computer graphics problems:

- Give reliable way to probe the global behavior of functions
- Lead naturally to robust, adaptive algorithms
- Several good libraries available on the internet

Affine arithmetic is a useful tool for interval methods

- AA more accurate than IA
- AA provides additional information that can be exploited
- AA locally more expensive than IA but globally more eficient
- AA has geometric flavor

Lots more to be done!

Some references

- Mudur–Koparkar, *IEEE CG&A*, 1984
  general philosophy
- Toth, *SIGGRAPH*, 1985
  ray tracing parametric surfaces
- Mitchell, *Graphics Interface*, 1990
  ray tracing implicit surfaces
- Suffern–Fackerell, *Computers & Graphics*, 1991
  interval methods in computer graphics
- Mitchell, *SIGGRAPH Course Notes*, 1991
  three applications

Some references

- Duff, *SIGGRAPH*, 1992
  implicit functions and constructive solid geometry
- Snyder, *SIGGRAPH*, 1992; also book
  interval analysis for computer graphics
- Gleicher–Kass, *Graphics Interface*, 1992
  intersection of parametric surfaces
- Barth–Lieger–Schindler, *The Visual Computer*, 1994
  ray tracing parametric surfaces
- Heidrich–Slusallek–Seidel, *ACM TOG*, 1998
  sampling procedural shaders
- Tupper, MSc thesis, Toronto, 1996; also *SIGGRAPH*, 2001
  plotting implicit relations with GraphEq