Computer Graphics

From mathematics …

$S = \{\mathbf{p} \in \mathbf{R}^3, f(\mathbf{p}) = 0\}$

$\mathbf{n} = -\nabla f(\mathbf{p})/|\nabla f(\mathbf{p})|$

… to the screen

E. Galin
Université Lyon 1

Computer Graphics

Core

Modeling

Ray Tracing

Meshing

Introduction

Visualization

$$S = \{\mathbf{p} \in \boldsymbol{R}^3 | f(\mathbf{p}) = 0\}$$

Polygonization [Araujo2015] converts model to large meshes

Direct **ray tracing** remains computationally **intensive**
Parallel implementation partially alleviates the problem

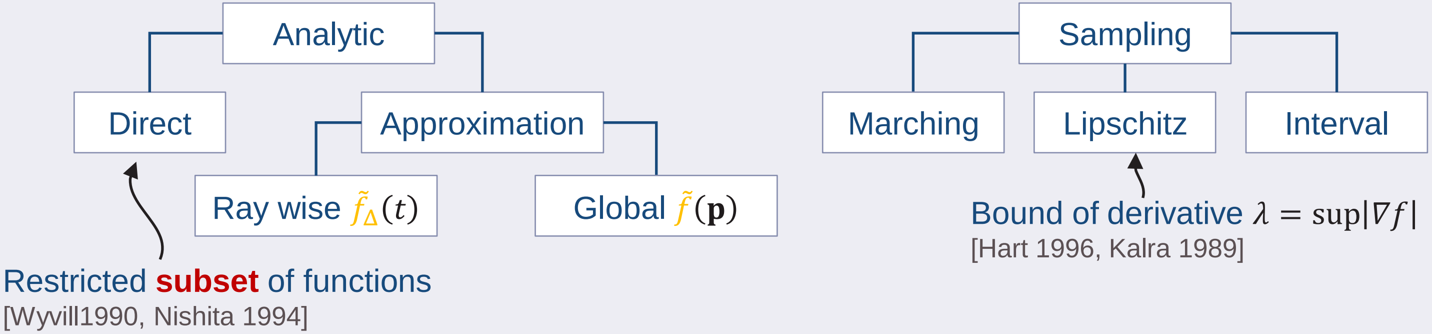
Ladybug
© Inigo Quilez

eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin

Problem statement

Solve $S \cap \Delta$ with $S = \{\mathbf{p} \in \boldsymbol{R}^3, f(\mathbf{p}) = 0\}$ and $\Delta = \{\delta(t), t \in [0, +\infty]\}$

Thus solve $f(t) = f \circ \delta(t) = 0$ over $[0, +\infty]$



J. Snyder. Interval analysis for computer graphics. *Computer Graphics*, **26** (2), SIGGRAPH, 1992
J. Hart. Sphere Tracing: A Geometric Method for the Anti aliased Ray Tracing of Implicit Surfaces. *The Visual Computer* **12**(10), 527-545,1996.

eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin

Analytic approaches

Analytic techniques

Algorithm

If f is algebraic of degree n, then $f \circ \delta(t)$ is a polynomial of degree n ← Can be solved **analytically** if $n \leq 4$

For a sphere $S = \{\mathbf{p}, (\mathbf{p}-\mathbf{c})^2 - r^2 = 0\}$ and $\delta(t) = \mathbf{o} + \mathbf{d}t$

$$d^2 t^2 + 2(o-c)\cdot t + (o-c)^2 - r^2 = 0$$

Blobs

f is the sum of compactly supported algebraic kernels

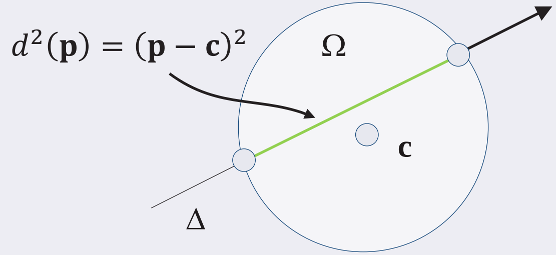
Then $f \circ \delta(t)$ is a **piecewise** polynomial of degree n [Wyvill1990]

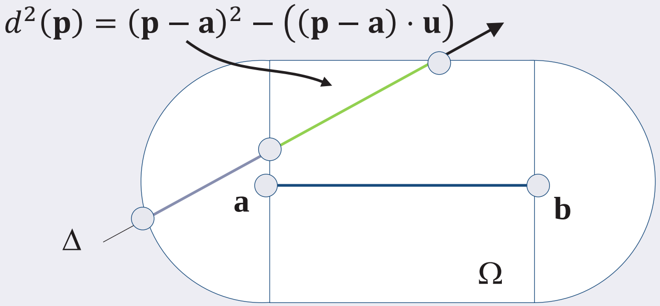
$$f(\mathbf{p}) = \sum_{i=0}^{n-1} g_i \circ d_i(\mathbf{p}) - T$$

Degree of g → $\sum$ upper bound

Falloff g_i of the squared distance d^2 → g_i

Distance to skeleton → $d_i(\mathbf{p})$





G. Wyvill. Ray-Tracing Soft Objects, Computer Graphics International, 469–476, 1990.

Université Claude Bernard Lyon 1
eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin

Approximation along the ray

Problem statement

In general, f and $f \circ \delta$ is not algebraic

Approximate $f \circ \delta$ along the ray

Piecewise Hermite polynomials $\tilde{f}_\Delta$ [Sherstyuk1999] and solve $\tilde{f}_\Delta = 0$

Set points as $t_i = \Omega_i \cap \Delta$

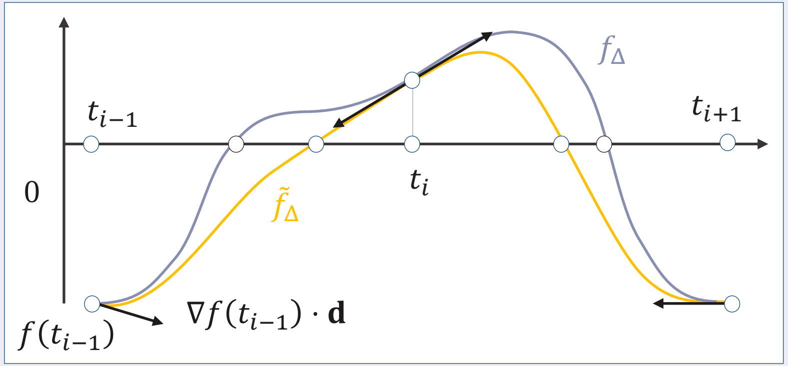
Approximate Ray Tracing

Sample ray Δ and with points t_i

Approximate $f \circ \delta$ by Hermite polynomial $\tilde{f}_\Delta$

Solve $\tilde{f}_\Delta = 0$

Values are $f(t_i)$ and derivatives set as $\nabla f(t_i) \cdot \mathbf{d}$



T. Nishita, E. Nakamae. A Method for Displaying Metaballs by using Bézier Clipping. *Computer Graphics Forum*, **13**(3), 271-280, 1994.

A. Sherstyuk. Fast Ray Tracing of Implicit Surfaces. *Computer Graphics Forum*, **18**(2), 139-147, 1999.

Université Claude Bernard Lyon 1

eric.galin@liris.cnrs.fr

http://liris.cnrs.fr/~egalin

Problem statement

Approximate f in $\mathbf{R}^3$ by a piecewise polynomial so that $f \circ \delta$ is algebraic
Decompose space into cells C_{ijk}
Approximate f by $\tilde{f}_{C_{ijk}}$ and solve $\tilde{f}_{C_{ijk}} \circ \delta = 0$ in cells C_{ijk}

Université Claude Bernard Lyon 1
eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin

Marching approaches

Ray marching

Algorithm

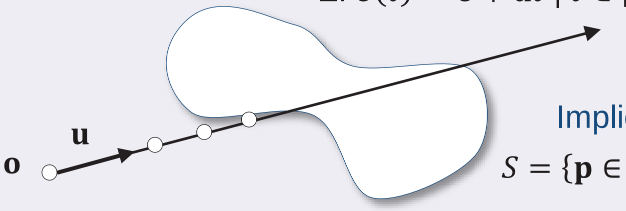
March along the ray at positions $\mathbf{p}_k$ with fixed steps ε

General algorithm that steps and evaluates f at points $\mathbf{p}$ along the ray with **fixed steps** [Perlin1989]

Ray Marching
Start from ray origin $t = 0$
At every step i
- $\mathbf{p}_i = o + t\mathbf{u}$
- If $f(\mathbf{p}_i) < 0$ then
  - Intersection found
- Otherwise step forward
  - $t = t + \varepsilon$

Ray Δ

Δ: $\boldsymbol{\delta}(t) = \mathbf{o} + \mathbf{u}t \mid t \in [0, \infty]$



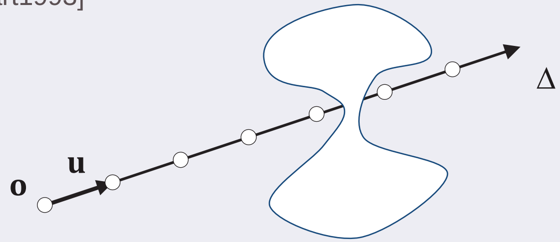
Implicit surface S

$S = \{\mathbf{p} \in \mathbf{R}^3 | f(\mathbf{p}) = 0\}$

No assumptions about the mathematical properties of f
Adapted for computing the intersection with fractal objects [Hart1993]
Computationally **intensive**

Caveats

Fixed precision ε is **slow** wherever S does not exist
Algorithm may **skip** S



J. Hart, D. Sandin, L. Kauffman. Ray Tracing Deterministic 3-D Fractals. ACM SIGGRAPH Computer Graphics, **23**(3), 1989.
K. Perlin, E. Hoffert. Hypertexture. *ACM SIGGRAPH Computer Graphics*, **23**(3), 1989.

Université Claude Bernard Lyon 1
eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin

Sphere Tracing

Algorithm

March along the ray at positions $\mathbf{p}_k$ with adaptive steps ε [Hart1996]
Simpler than the **robust** Lipschitz techniques with **global** constants [Kalra1989]

Lipschitz functions

$$\exists\lambda > 0\ \ \forall(\mathbf{p},\mathbf{q}) \in \mathbf{R}^3 \times \mathbf{R}^3\ |f(\mathbf{p}) - f(\mathbf{q})| \leq \lambda|\mathbf{p}-\mathbf{q}|$$

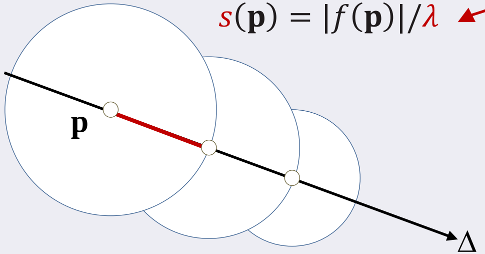
Exclusion criterion: $|f(\mathbf{p})|/\lambda$ is a signed distance bound to S

$$\forall\mathbf{p} \in \mathbf{R}^3\ B(\mathbf{p},|f(\mathbf{p})/\lambda|) \cap S = \emptyset$$

Global bound λ over $\mathbf{R}^3$

Sphere Tracing [Hart1996]
Start from ray origin $t = 0$
While $t < \infty$
  $\mathbf{p} = o + t\mathbf{u}$
  Compute $v = f(\mathbf{p})$
  If $v < 0$ then
    Intersection found
  Otherwise step forward
    $t = t + |v|/\lambda$

$s(\mathbf{p}) = |f(\mathbf{p})|/\lambda$



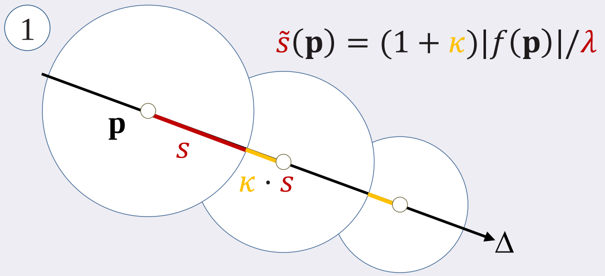
Sphere tracing

D. Kalra, H. Barr. Guaranteed Ray Intersections with Implicit Surfaces. *ACM SIGGRAPH Computer Graphics*, **23**(3), 1989.
J. Hart. Sphere Tracing: A Geometric Method for the Antialiased Ray Tracing of Implicit Surfaces. *The Visual Computer* **12**(10), 527-545,1996.

Université Claude Bernard Lyon 1
eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin

Improvements

Overstepping allows for slightly larger steps [Keinert2014]
Steps still **adapt** with a **global** constant λ
Need to **step back** in some cases

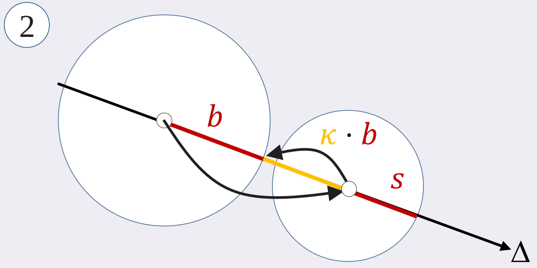


Improved sphere tracing

Improved Sphere Tracing
Start from ray origin $t = 0$ and $b = 0$
While $t < \infty$
- $\mathbf{p} = o + t\mathbf{u}$
- Compute $v = f(\mathbf{p})$ step $s = |v|/\lambda$
- If $s < \kappa \cdot b$
  - $t = t - \kappa \cdot b$ and $b = 0$ (2)
- Otherwise
  - If $v < 0$ then intersection found
  - $t = t + (1+\kappa)s$ and $b = s$ (1)

Tricky ordering of instructions

Improved step — Save safe step



Step back configuration

Larger values of κ increase overstep but also step back
Experiments suggest $\kappa \approx 0.2$

LIRIS CNRS Université Claude Bernard Lyon 1
eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin

Acceleration

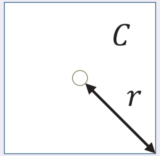
Octrees

Surface embedding structure

Octrees [Hart1996] define a hierarchy of cells inside, outside and **straddling** S
Cells may store a **local** Lipschitz constant λ

Cell exclusion criterion
$$|f(\mathbf{c})| \geq \lambda(C)\, r$$

Lipschitz constant in $C \supset S(\mathbf{c}, r)$

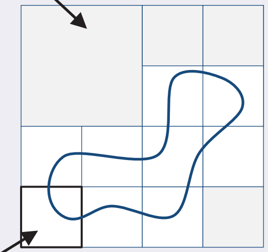


Octree construction
Start with cubic cell C
If $|f(\mathbf{c})| \geq \lambda(C)\, r$
    Mark C as in or out (empty)
Otherwise C may be straddling
    If C is a terminal cell
        Store the **local** constant $\lambda(C)$
    Otherwise
        Process sub cells C_k
        If 8 sub cells are empty
            Set C as empty

Empty space acceleration

Sphere tracing acceleration

Octree simplification



Ray marching or sphere tracing should be performed with $\lambda(C)$ on **sub intervals** $\Delta \cap O$

J. Hart. Sphere Tracing: A Geometric Method for the Anti aliased Ray Tracing of Implicit Surfaces. *The Visual Computer* **12**(10), 527-545,1996.

eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin

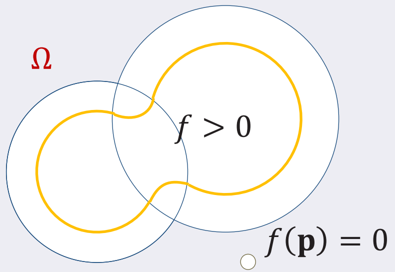
Bounding volumes

Compactly supported hierarchical models

BlobTrees [Wyvill1999] built with compactly supported nodes : $f = 0$ if $\mathbf{p} \notin \Omega$
Nodes can include a volume $V \supset \Omega$
f remains always correctly evaluated

Node evaluation
If $\mathbf{p} \notin V$ return 0
Otherwise compute $f(\mathbf{p})$

Avoid expensive evaluation of subtree f

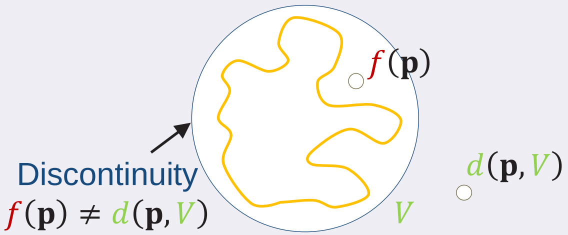


Signed distance fields

Nodes are not compactly supported
Bounding volumes V can replace the complex evaluation of $f(\mathbf{p})$ with simpler distance $d(\mathbf{p}, V)$
Simpler distance should satisfy $d(\mathbf{p}, V) > d(\mathbf{p})$
The new function $\tilde{f}$ is **piecewise** Lipschitz, not valid for **modeling**, sufficient for **sphere tracing**

Node evaluation
If $\mathbf{p} \notin V$ return $d(\mathbf{p}, V)$
Otherwise compute $f(\mathbf{p})$

Piecewise lower distance bound



B. Wyvill, A. Guy, E. Galin. Extending the CSG-Tree. *Computer Graphics Forum*. **18** (4), 149-158, 1999

Université Claude Bernard Lyon 1
eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin

Interval analysis

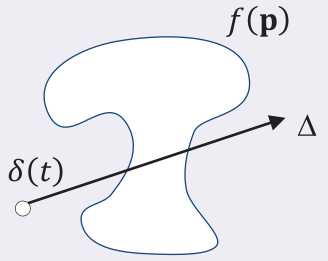
Interval analysis

Root finding algorithm

Computing $S \cap \Delta$ is the same as finding if $f \circ \delta$ has roots over an interval I [Snyder 1992]
Compute embedding interval $f(\mathrm{I})$, if $0 \notin f(\mathrm{I})$ then no root exists over I, otherwise perform bisection

Image of an interval by a function $\mathrm{E} \supset f(\mathrm{I})$

bool Solve($\mathrm{I} = [a, b]$)
If $0 \notin f(\mathrm{I})$ return false
Else
    Set $c = (a + b)/2$
    If $|b - a| < \varepsilon$ return true
    if Solve($[a, c]$) return true
    return Solve($[c, b]$)



Fundamentals

An interval $\mathrm{I} = [a, b] = \{x \in \mathbf{R}, a \leq x \leq b\}$
Computing the image of an interval by a function $f(\mathrm{I})$ is complex
Inclusion : find an embedding interval E such that $\mathrm{E} \supset f(\mathrm{I})$
Rewrite operators and function for the evaluation of f for intervals

Operators

$\mathrm{A} + \mathrm{B} = [\underline{a} + \underline{b}, \overline{a} + \overline{b}]$

$\mathrm{A} \times \mathrm{B} = [\underline{a} \times \underline{b}, \overline{a} \times \overline{b}]$ if $\underline{a} > 0$ and $\underline{b} > 0$

Functions

$f(\mathrm{A}) = [f(\underline{a}), f(\overline{a})]$ if f monotonous

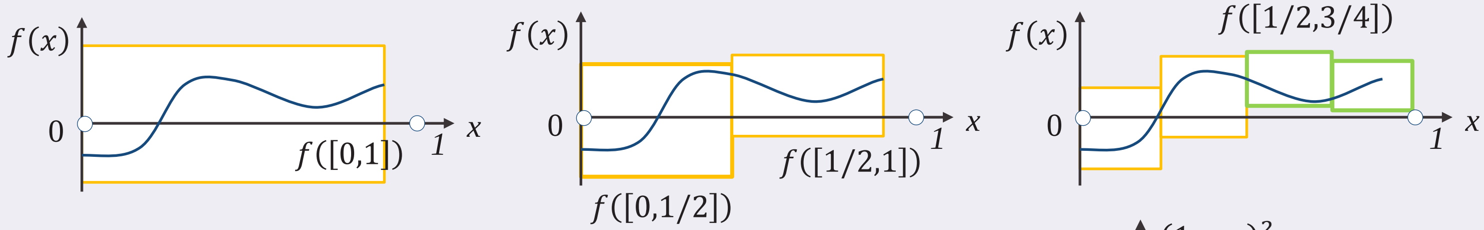
$\sqrt{A} = [\sqrt{\underline{a}}, \sqrt{\overline{a}}]$ if $\underline{a} > 0$

J. Snyder. Interval analysis for computer graphics. *Computer Graphics*, **26** (2), SIGGRAPH, 1992

Université Claude Bernard Lyon 1
eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin

Analysis

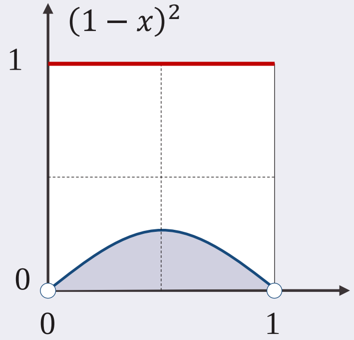
Recursion occurs if $0 \notin f(\mathrm{I})$



Limitations

The embedding interval $\mathrm{E} \supset f(\mathrm{I})$ can be a large approximation

With $x \in [0,1]$, we have $f(x) = (1-x)^2 \in [0,1/4]$
With intervals $\mathrm{I} = [0,1]$, then $1 - \mathrm{I} = [0,1]$ and $\mathrm{I}^2 = [0,1]$



Université Claude Bernard Lyon 1

eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin