

Computer Graphics

From mathematics ...



... to the screen

E. Galin
Université Lyon 1

Computer Graphics

Core
Modeling
Ray Tracing
Meshing

Computer Graphics

Introduction

Etat de l'art

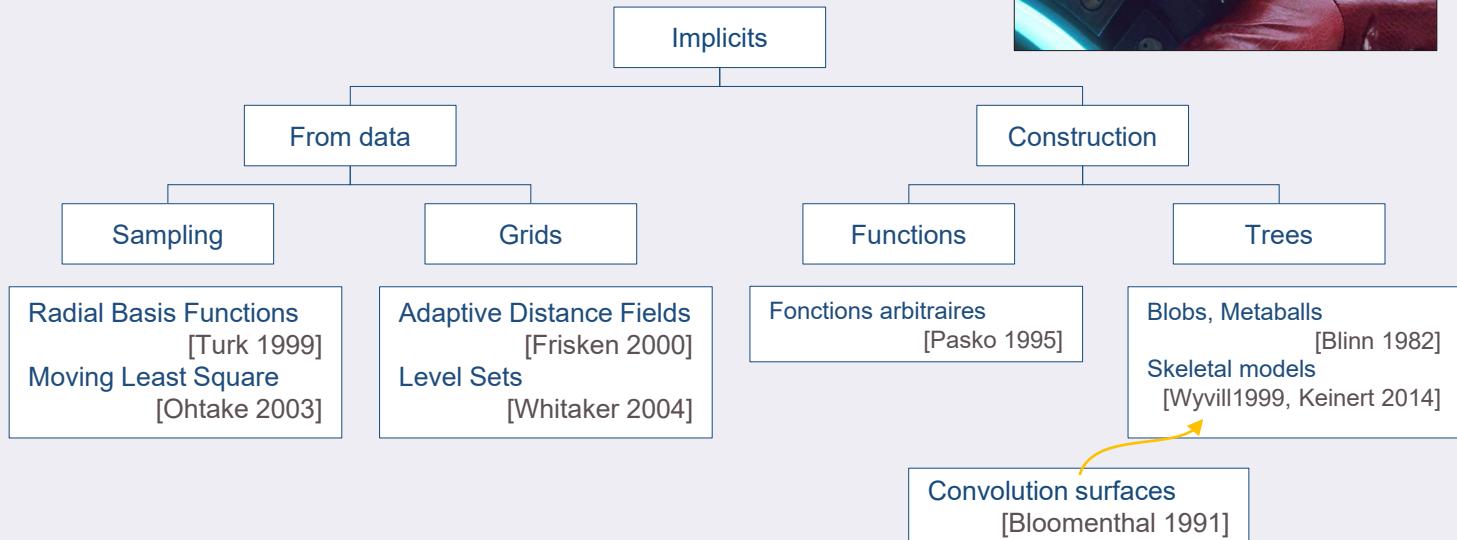
Introduction

Blobs
Distance fields
Geometry
Optimization
Convolution

Problème

Construction de f
Modélisation hiérarchique
Combinaison de formes

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



James Blinn. A Generalization of Algebraic Surface Drawing. *ACM Transactions on Graphics*, 1:235, 1982.

A. Pasko, V. Adzhiev, A. Sourin, and V. Savchenko. Function representation in geometric modeling: concepts, implementation and applications. *The Visual Computer*, 2(8), 429-446, 1995.

B. Wyvill, A. Guy and E. Galin. Extending the CSG Tree (Warping, Blending and Boolean Operations in an Implicit Surface Modeling System). *Computer Graphic Forum*, 18(2), 149-158, 1999.

Computer Graphics

Algebraics

Surfaces algébriques

Introduction

Blobs

Distance fields

Geometry

Optimization

Convolution

Formulation

Équations polynomiales de degré n

Calcul de $f(\mathbf{p})$ rapide pour un degré faible

Calcul analytique de $\nabla f(\mathbf{p})$

Certaines formes comme les cyclides [Foufou2004] très utiles

Manque de contrôle et de variété de formes

$$f(\mathbf{p}) = f(x, y, z) = \sum_{0 \leq i+j+k \leq n} a_{ijk} x^i y^j z^k$$



Smoothed Kummer Surface



eric.galin@liris.cnrs.fr

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

J. Blinn. A Generalization of Algebraic Surface Drawing, *IEEE Transactions on Graphics*, 1(3), 235 – 256, 1982

S. Foufou, L. Garnier. Dupin cyclides as quadrics blends for shape modeling. *Computer Graphics Forum*, 23(3), 321 – 330, 2004

Computer Graphics

Blobs

Blobs

Introduction
Blobs
Distance fields
Geometry
Optimization
Convolution

Définition

Objet construit à partir d'éléments définis à partir de squelettes et se raccordant entre eux
Iso surface d'une somme de fonctions potentiel décroissante selon la distance à sommets

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

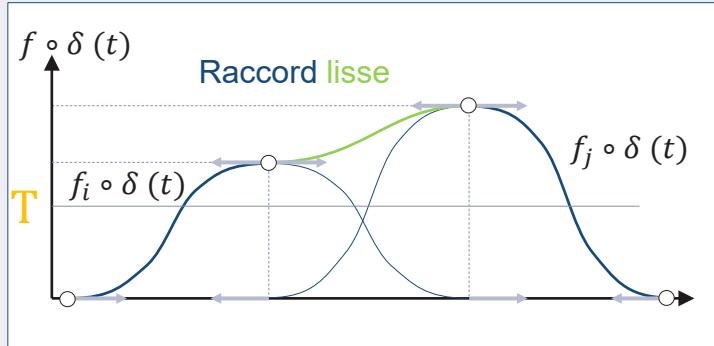
$$f = \sum_i f_i(\mathbf{p})$$

$$f_i(\mathbf{p}) = g_i \circ d_i(\mathbf{p})$$

Fonction d'atténuation

Distance au point \mathbf{c}_i

Un élément est caractérisé par un point \mathbf{c}_i , sa fonction distance d_i à \mathbf{c}_i , une fonction d'atténuation d_i



J. Blinn. A Generalization of Algebraic Surface Drawing. *ACM Transactions on Graphics*. 1 (3): 235–256, 1982
B. Wyvill. Data Structure for Soft Objects. *The Visual Computer*, 2(4): 227 – 234, 1986

Fonctions d'atténuation

Introduction
Blobs
Distance fields
Geometry
Optimization
Convolution

Fonctions potentiel

Distribution de potentiel autour du squelette avec certaines propriétés

Symétrie $g(d) = g(1 - d)$, centre $g(1/2) = 1/2$

Dérivabilité aux bord $g^n(0) = 0$ et $g^n(1) = 0$

Carré de la distance r^2

$$g(d) = 2d^3 - 3d^2 + 1$$

Hermite

$$g(d) = (1 - d^2)^n \quad n \geq 2$$

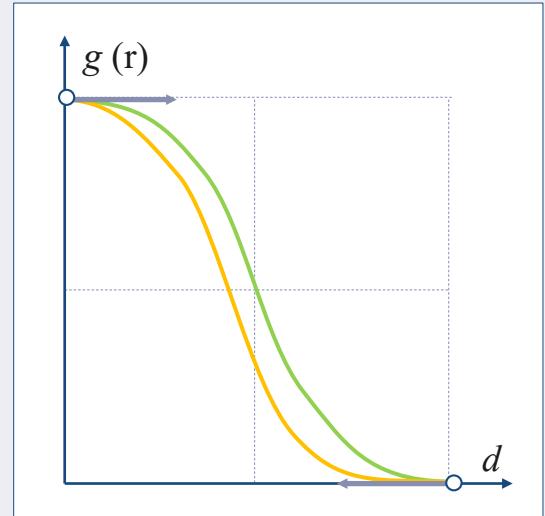
Wyvill

$$g(d) = (1 - d^2)^2 (1 - 4/9 d^2)$$

Polynomiale

Régularité	Symétrie	Centre	Carré
Hermite	C^1	✓	✓
Wyvill	C^1	✗	✓
Polynomiale	C^{n-1}	✗	✓

Hermite	C^1	✓	✓
Wyvill	C^1	✗	✓
Polynomiale	C^{n-1}	✗	✓



Arbre de construction

Introduction
Blobs
Distance fields
Geometry
Optimization
Convolution

Construction

Combinaison hiérarchique [Wyvill1999]

Primitives aux feuilles, squelettes complexe : courbes, cube, cône, cylindre, polyèdre

Opérateurs aux nœuds : mélange, union, intersection, différence, déformations

Opérateurs de composition

Union, intersection, différence

$$f_{A \cup B}(\mathbf{p}) = \max(f_A, f_B)$$

$$f_{A \cap B}(\mathbf{p}) = \min(f_A, f_B)$$

$$f_{A-B}(\mathbf{p}) = \min(f_A, 2T - f_B)$$

Les fonctions min et max sont C⁰

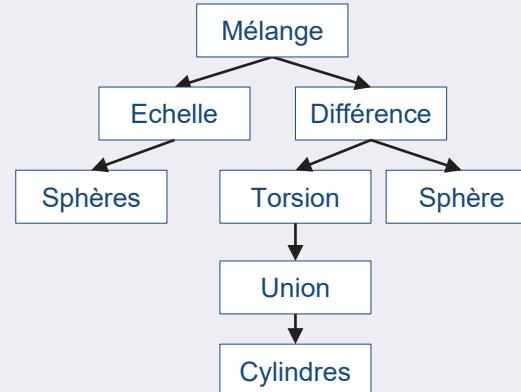
Fonctions plus régulières [Pasko1995]

$$f_{A \cup B}(\mathbf{p}) = f_A + f_B + (f_A^2 + f_B^2)^{1/2}$$

Mélange et mélange généralisé

$$f_{A \circ B}(\mathbf{p}) = f_A + f_B$$

$$f_{A * B}(\mathbf{p}) = (f_A^n + f_B^n)^{1/n} \quad n \geq 1$$



eric.galin@liris.cnrs.fr

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

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

Computer Graphics

Distance fields

Preliminaries

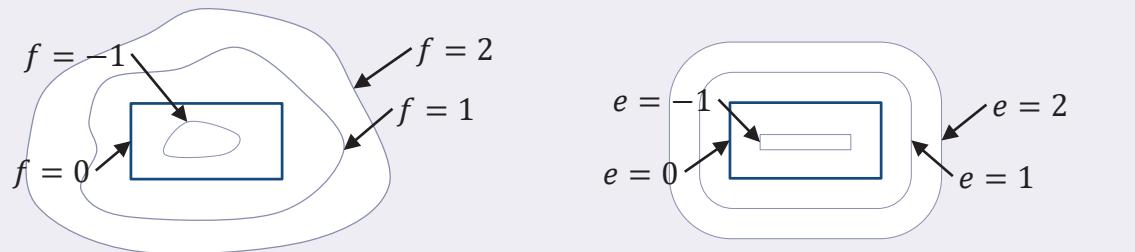
Introduction
Blobs
Distance Fields
Geometry
Optimization
Convolution

Interpretation of distance field

In general, the signed function f is **not** the Euclidean distance to the surface S

Recall that if f is λ -Lipschitz, then f/λ is a **lower** signed **Euclidean distance bound** to S [Hart1996]

$$\forall \mathbf{p} \in \mathbb{R}^3 |f(\mathbf{p})|/\lambda < e(\mathbf{p}) \text{ with } e(\mathbf{p}) = d(\mathbf{p}, S)$$



Euclidean distance

If f is the signed Euclidean distance to the surface S , then $|\nabla f(\mathbf{p})| = 1$

Eikonal equation

Therefore, is $1 - \text{Lipschitz}$

Primitives

Introduction
Blobs
Distance Fields
Geometry
Optimization
Convolution

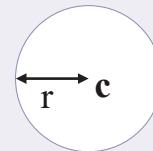
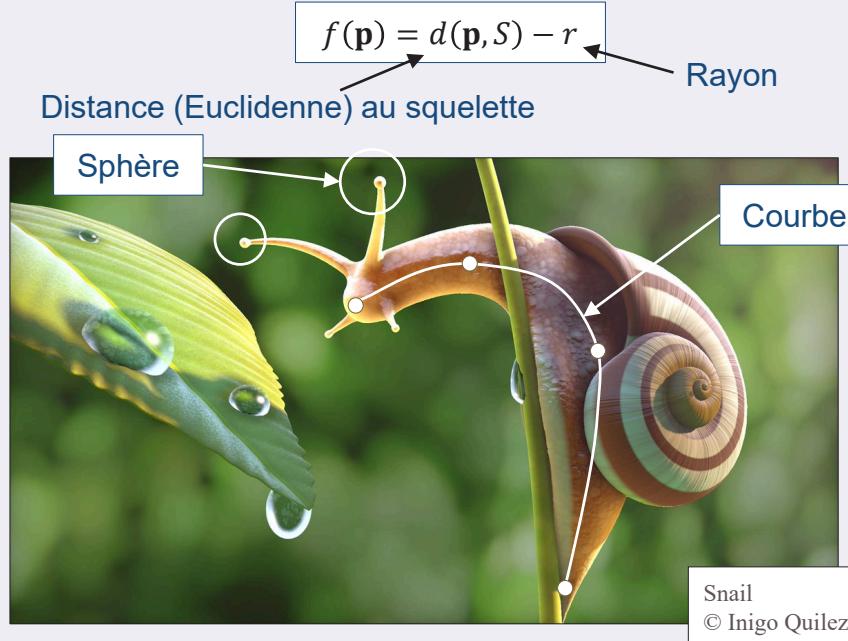
Distance à un squelette

Convention $f(\mathbf{p}) < 0$ si \mathbf{p} dans l'objet, $f(\mathbf{p}) > 0$ dehors

Tore, disque arrondi, barre arrondie, sphère, sphère le long d'une courbe

Vidéo

Vidéo



$$f(\mathbf{p}) = ||\mathbf{p} - \mathbf{c}|| - r$$

Primitives polygonales (boîte, pyramide) : intersections de demi espaces...

$$d(\mathbf{p}, \pi) = (\mathbf{p} - \mathbf{c}) \cdot \mathbf{n}$$

Opérateurs de combinaison

Introduction
Blobs
Distance Fields
Geometry
Optimization
Convolution

Opérateurs booléens

Union, intersection, différence

$$f_{A \cup B}(\mathbf{p}) = \min(f_A, f_B)$$

$$f_{A \cap B}(\mathbf{p}) = \max(f_A, f_B)$$

$$f_{A-B}(\mathbf{p}) = \max(f_A, -f_B)$$

Mélange

Dans le domaine des Blobs, on calcule $f_A + f_B$ simplement [Wyvill1999]
Modèle plus complexe pour les distances signées

$$f_{A \cup^* B}(\mathbf{p}) = \min(f_A, f_B) - g(f_A, f_B)$$

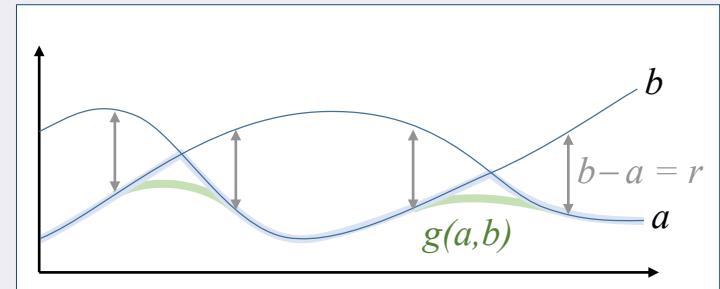
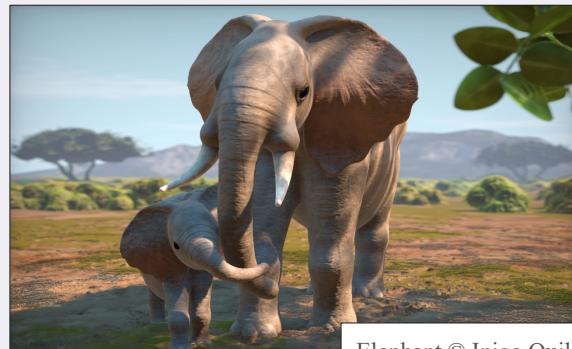
Union

Offset

$$\text{Correction } g(a, b) = \frac{1}{6} r h^3$$

avec $h = \max(r - |a - b|, 0)/r$

Rayon de mélange



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

Transformations affines

Introduction
Blobs
Distance Fields
Geometry
Optimization
Convolution

Principe

En modélisation implicite, on déforme l'espace par la transformation inverse

Translation, rotation, homothétie

$$f_{T(A)}(\mathbf{p}) = f_A \circ (\mathbf{p} - \mathbf{t})$$

Translation inverse

$$f_{\omega(A)}(\mathbf{p}) = 1/\|J_{\omega}^{-1}\| f_A \circ \omega^{-1}$$

Norme du Jacobien

$$f_{R(A)}(\mathbf{p}) = f_A \circ R^{-1}(\mathbf{p})$$

Rotation inverse
 $R^{-1}(\theta) = R(-\theta)$

$$f_{S(A)}(\mathbf{p}) = 1/s f_A \circ (\mathbf{p}/s)$$

Homothétie inverse

Les compositions de transformations ne
sont **pas commutatives**

Matrices de rotation

Différentes matrices selon les repères [Shirley]

$$R_x(\theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & c & -s \\ 0 & s & c \end{pmatrix}$$

où $c = \cos \theta$ et $s = \sin \theta$

$$R_u(\theta) = \begin{pmatrix} x^2a + c & xy a - sz & xz a + sy \\ xy a + sz & y^2a + c & yz a - sx \\ xz a - sy & yz a + sx & z^2a + c \end{pmatrix}$$

où $a = 1 - \cos \theta$



eric.galin@liris.cnrs.fr

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

P Shirley. Fundamentals of Computer Graphics. Third Edition. AK Peters

Déformation

Introduction
Blobs
Distance Fields
Geometry
Optimization
Convolution

Définition

Déformation d'une région de l'espace
Torsion, écrasement [Barr1984]

$$f_{\omega(A)}(\mathbf{p}) = 1/\|\mathbf{J}_{\omega}^{-1}\| f_A \circ \omega^{-1}$$

La fonction ω doit être **inversible**
Expression du **Jacobien** de ω^{-1} complexe

$$\nabla f_{\omega(A)}(\mathbf{p}) = (\mathbf{J}_{\omega}^{-1})^t(\mathbf{p}) \nabla f_A(\mathbf{p})$$



Perturbation par bruit

Déplacement stochastique $\delta(\mathbf{p})$
Fractional Brownian Motion

$$f_{\omega(A)}(\mathbf{p}) = f_A \circ (\mathbf{p} - \delta(\mathbf{p}))$$

Bruit **n** ou turbulence **t**

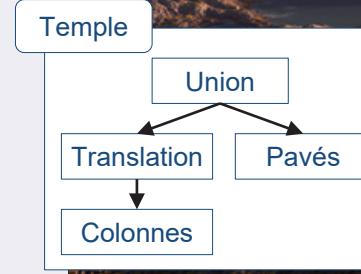
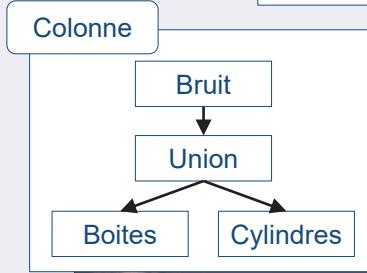
Composition

Introduction
Blobs
Distance Fields
Geometry
Optimization
Convolution

Formes complexes

Construction de **formes complexes** par assemblage hiérarchique de primitives
Placement d'**instances** et composition dans le graphe de scène

Graphe de scène \Leftrightarrow Composition de fonctions \Leftrightarrow Appels



Temple © Inigo Quilez

Computer Graphics

Geometric primitives

Distance à un segment

Introduction
Blobs
Distance Fields
Geometry
Optimization
Convolution

Distance à une droite

Pythagore dans le triangle apq

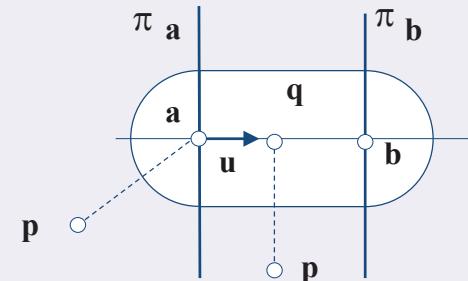
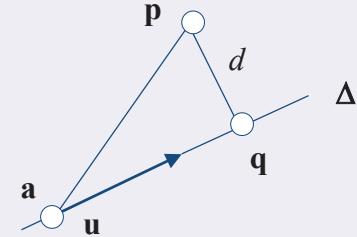
$$d^2(p, \Delta) = (\mathbf{p} - \mathbf{a})^2 - ((\mathbf{p} - \mathbf{a}) \cdot \mathbf{u})^2$$

Carré de la distance

Distance à un segment

Calcul de $l = (\mathbf{p} - \mathbf{a}) \cdot \mathbf{u}$ où $\mathbf{u} = (\mathbf{b} - \mathbf{a}) / |(\mathbf{b} - \mathbf{a})|$

$$\begin{aligned} & (\mathbf{p} - \mathbf{a})^2 \text{ si } l < 0 \\ d^2(\mathbf{p}, [\mathbf{ab}]) &= (\mathbf{p} - \mathbf{a})^2 - ((\mathbf{p} - \mathbf{a}) \cdot \mathbf{u})^2 \text{ si } 0 < l < |\mathbf{b} - \mathbf{a}| \\ & (\mathbf{p} - \mathbf{b})^2 \text{ sinon} \end{aligned}$$

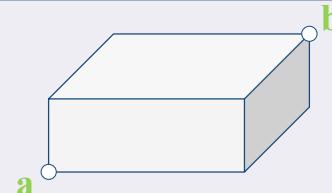
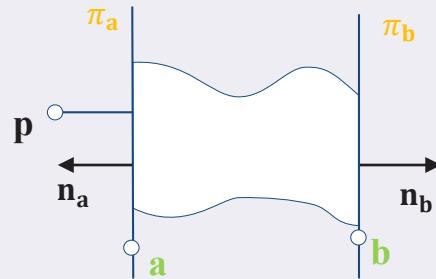


Distance à une boîte

Introduction
Blobs
Distance Fields
Geometry
Optimization
Convolution

Distance Euclidienne

On utilise la distance entre \mathbf{p} et deux plans parallèles



$$d(\mathbf{p}, \pi_a \cap \pi_b) = \begin{cases} 0 & \text{si } a_x < \mathbf{p}_x < b_x \\ a_x - \mathbf{p}_x & \text{si } \mathbf{p}_x < a_x \\ \mathbf{p}_x - b_x & \text{si } \mathbf{p}_x > b_x \end{cases}$$

$$d^2(\mathbf{p}, B) = d^2(\mathbf{p}, S_x) + d^2(\mathbf{p}, S_y) + d^2(\mathbf{p}, S_z)$$

Distance signée

Soit $\mathbf{c} = (\mathbf{a} + \mathbf{b})/2$ le centre, $\tilde{\mathbf{p}} = \mathbf{p} - \mathbf{c}$ et $\mathbf{d} = (\mathbf{b} - \mathbf{a})/2$ la diagonale
On tire parti de la symétrie en définissant $\mathbf{q} = \text{abs}(\mathbf{p}) - \mathbf{d}$

$$d(\mathbf{p}, B) = \min(\max(q_x, q_y, q_z), 0) + |\max(\mathbf{q}, \mathbf{0})|$$

Annotations below the equation:

- 0 si $\mathbf{q} \notin B$
- Maximum de $q_i < 0$ si $\mathbf{q} \in B$
- Norme Euclidienne
- 0 si $\mathbf{q} \in B$
- Vecteur nul

Distance à un cercle ou un disque

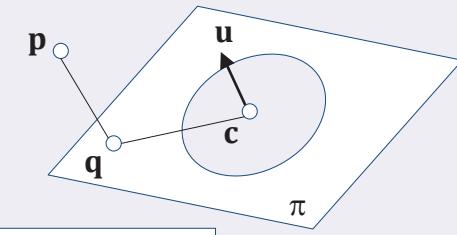
Introduction
Blobs
Distance Fields
Geometry
Optimization
Convolution

Cercle

Par construction $d(\mathbf{p}, C)^2 = d(\mathbf{p}, \pi)^2 + ((\mathbf{c} - \mathbf{q}) - r)^2$ ou \mathbf{q} projeté de \mathbf{p}

```
double Circle::R(const Vector& p) const {
    Vector n = p - c;
    double h = n * u;
    double d = n * n;
    double y = d - h * h;
    y = r - sqrt(y);
    return sqrt(y * y + h * h);
}
```

// Distance to center
// Radial distance
// Distance to **circle**



if ($y < r^r$) { return z; } // Distance to **disc**

Disque

On teste si $|\mathbf{c} - \mathbf{q}| < r$: dans ce cas où l'on calcule la distance au plan π

Distance à une courbe

Introduction
Blobs
Distance Fields
Geometry
Optimization
Convolution

Distance d'un point à une courbe

Soit Γ une courbe d'équation paramétrée $\mathbf{c}(t)$, $t \in [0,1]$

$$d^2(\mathbf{p}, \Gamma) = \min_{t \in [0,1]} (\mathbf{c}(t) - \mathbf{p})^2$$

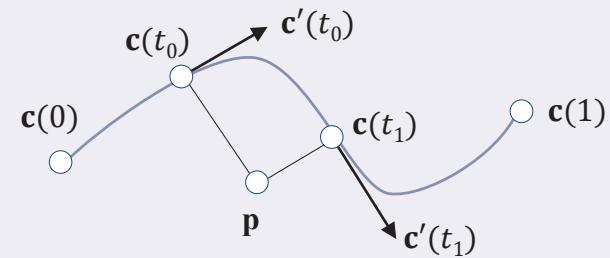
Polynôme de degré n

Recherche des racines de l'équation

$$d^2(\mathbf{p}, \Gamma)' = 0 \Leftrightarrow (\mathbf{c}(t) - \mathbf{p}) \cdot \mathbf{c}'(t) = 0$$

Équation de degré $2n - 1$

Coût relatif de $\approx 1:20$ pour une courbe quadrique par rapport à une cubique



Distance à des formes polygonales

Introduction
Blobs
Distance Fields
Geometry
Optimization
Convolution

Polygone

Minimum de la distance aux segments et test d'intérieur, coût en $O(n)$

$$d^2(\mathbf{p}, \Pi) = s(\mathbf{p}) \min_{i \in [0, n-1]} d(\mathbf{p}, \mathbf{e}_i)$$

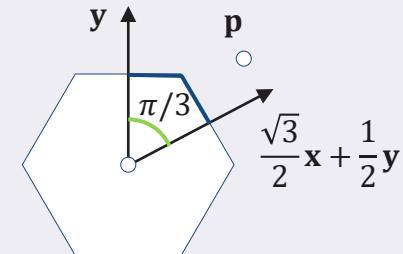
Signe > 0 si extérieur à Π , < 0 sinon Arête $\mathbf{e}_i = \mathbf{c}_i \mathbf{c}_{i+1}$

```
double Polygon2::Signed(const Vector2& p) const
{
    double r = sqrt(Contour(p)); // Distance to contour
    return Inside(p) ? -r : r;
}
```

Hexagone, polygones réguliers

Accélération grâce aux symétries, coût en $O(1)$

Distance à deux droites ou à un sommet



eric.galin@liris.cnrs.fr

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

Computer Graphics

Performance and optimisation

Optimisations

Introduction
Blobs
Distance Fields
Geometry
Optimization
Convolution

Trigonometrie

Calcul de $\cos(2k\pi/n)$ et $\sin(2k\pi/n)$ dans une boucle

Set $\mathbf{R}_\theta = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$ with $\theta = 2\pi/n$

Set $\mathbf{u} = (1,0)$

For i from 0 to n

Computations with \mathbf{u}

$\mathbf{u} \leftarrow \mathbf{R}_\theta \cdot \mathbf{u}$

Stores $\cos(2k\pi/n)$ et $\sin(2k\pi/n)$

Cost $4 \times$ and $2 \pm$ instead of trigonometrics



eric.galin@liris.cnrs.fr

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

Computer Graphics

Convolution surfaces

Convolution surfaces

Introduction
Blobs
Distance Fields
Geometry
Optimization
Convolution

Définition

Convolution d'une fonction de géométrie g avec un noyau k

A l'origine, g est une fonction caractéristique et k le noyau Gaussien [Bloomenthal1991]

$$f(\mathbf{p}) = g * k(\mathbf{p}) = \int_{\mathbb{R}^3} g(\mathbf{q})k(|\mathbf{p} - \mathbf{q}|) d\mathbf{q}$$

Géométrie, squelette

Noyau $k(x) = e^{-x^2}$

Support **infini**, coûteux

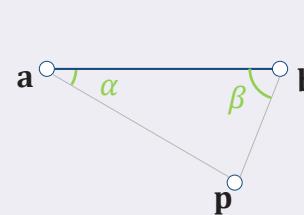
Autres noyaux possibles pour permettre un calcul analytique [Sherstyuk 1999]

$$k(x) = \frac{1}{1 + \sigma^2 x^2}$$

Fonction de Cauchy [Sherstyuk 1998]

$$k(x) = \frac{1}{x^3}$$

Puissance inverse [Cani 2001, Hornus 2003]



$$f(\mathbf{p}) = \frac{\sin \alpha - \sin \beta}{d^2(\mathbf{p}, \mathbf{ab})}$$

J. Bloomenthal, K. Shoemake. Convolution surfaces. *Computer Graphics* 25(4), 251-255. 1991.

A. Sherstyuk. Kernel functions in convolution surfaces: a comparative analysis. *The Visual Computer* 15(4), 171-182, 1999.

J. McCormack, A. Sherstyuk. Creating and rendering convolution surfaces. *Computer Graphics Forum* 17(2), 113-120, 1998.

Hornus, S., Angelidis, A., Cani, M.-P., Implicit modelling using subdivision curves. *The Visual Computer* 19(2-3), 94-104, 2003.