

# Computergraphik II

Winter 2012/2013

## 6 Implizite Flächen

Versionsdatum: 15. November 2012



## 6 Implizite Flächen ...



**Ziel:** Modellierung geschlossener, glatter (komplexer) Objekte

### Vorteile impliziter Flächen:

- einfache Modellierung (keine zusammengesetzten NURBS-Flächen)
- Modellierung geschlossener, glatter Körper wie Sphären; Beispiel:

$$K = \{(x, y, z) \in \mathbb{R}^3 : x^2 + y^2 + z^2 = r^2\}, \quad \text{Radius: } r > 0$$

- z.T. einfache Schnitt- und Abstandsberechnungen (z.B für Kollision)
- Anwendung als Bounding Volume

### Nachteile impliziter Flächen:

- Rastergraphik: Polygonalisierung impliziter Flächen aufwendig
- Geometrische Größen z.T. schwer ermittelbar

### Bemerkung:

- Die Begriffe *implizite Fläche* und *Isofläche* werden hier synonym verwendet
- Im folgenden stellen Graphiken immer den 2D-Fall dar (*implizite Kurven*)





## Beispiel: Kugel als parametrische Fläche

**Parametrisch** Kugel mit Radius 1 im Ursprung.  
Formulierung als Funktion in den Parametern  $\vartheta, \varphi$

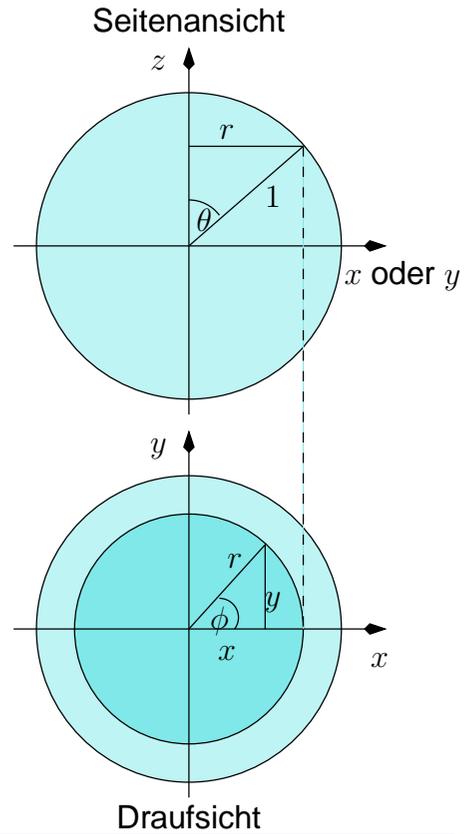
$$\mathbf{F}(\vartheta, \varphi) = \begin{pmatrix} \sin \vartheta \cos \varphi \\ \sin \vartheta \sin \varphi \\ \cos \vartheta \end{pmatrix}, \quad \vartheta \in [0, \pi], \quad \varphi \in [0, 2\pi]$$

## Seitenansicht und Draufsicht der Kugel

$$r = 1 \cdot \sin \vartheta \quad x = r \cdot \cos \varphi \Rightarrow x = \sin \vartheta \cdot \cos \varphi$$

$$z = 1 \cdot \cos \vartheta \quad y = r \cdot \sin \varphi \Rightarrow y = \sin \vartheta \cdot \sin \varphi$$

( $\vartheta \hat{=}$  Breitengrad,  $\varphi \hat{=}$  Längengrad, Geographie:  $0^\circ =$  Äquator, hier:  $0^\circ =$  Nordpol)



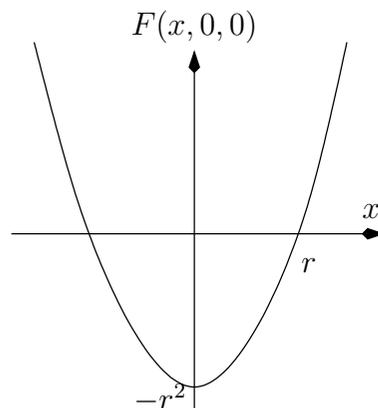
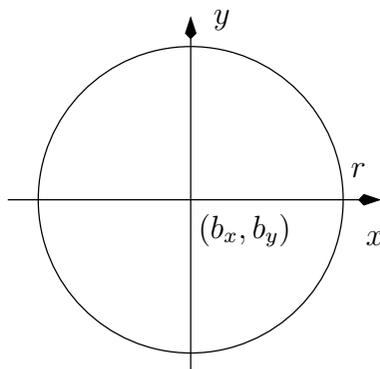
## Beispiel: Kugel als implizite Fläche

**Implizit Definition** via „Auswertefunktion“  $F : \mathbb{R}^3 \rightarrow \mathbb{R}$

P ist genau dann Oberflächenpunkt, wenn  $F(\mathbf{P}) = 0$

**Kugel** mit Radius r im Punkt B.

$$\{\mathbf{P} \in \mathbb{R}^3 : F(\mathbf{P}) = 0\}, \quad \text{mit } F(\mathbf{P}) = (p_x - b_x)^2 + (p_y - b_y)^2 + (p_z - b_z)^2 - r^2$$

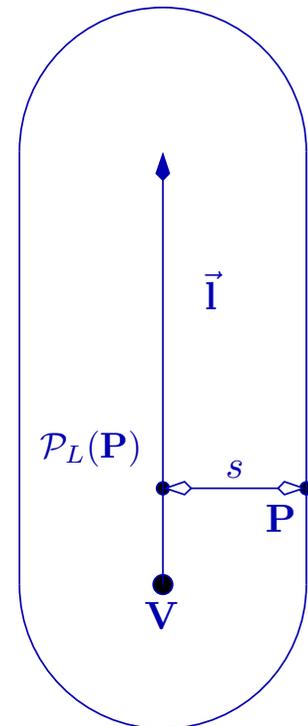




### Definition:

Implizite Geometrien werden definiert durch:

Implizite Fläche	Kapsel
Basisgeometrie $B$	Strecke $L$ $\mathbf{L}(\alpha) = \mathbf{V} + \alpha \vec{\mathbf{I}}, \alpha \in [0, 1]$
Distanzfunktion $d_B(\mathbf{P})$ Abstand von $\mathbf{P}$ zur Basis- geometrie	$d_L(\mathbf{P}) = \min\{\ \mathbf{P} - \mathbf{Q}\  : \mathbf{Q} \in L\}$ (Abstand von $\mathbf{P}$ zur $L$ )
Schwellwert/Isowert: $s > 0$	$s > 0$
Geometrie: $M_s = \{\mathbf{P} \in \mathbb{R}^3 : d_B(\mathbf{P}) = s\}$	Kapsel: $M_s = \{\mathbf{P} \in \mathbb{R}^3 : d_L(\mathbf{P}) = s\}$



### 6.1.1 Kapsel ...



#### Abstandsberechnung

Ermittlung der Projektion  $\mathcal{P}_L(\mathbf{P})$  von  $\mathbf{P}$  auf  $L$ , also

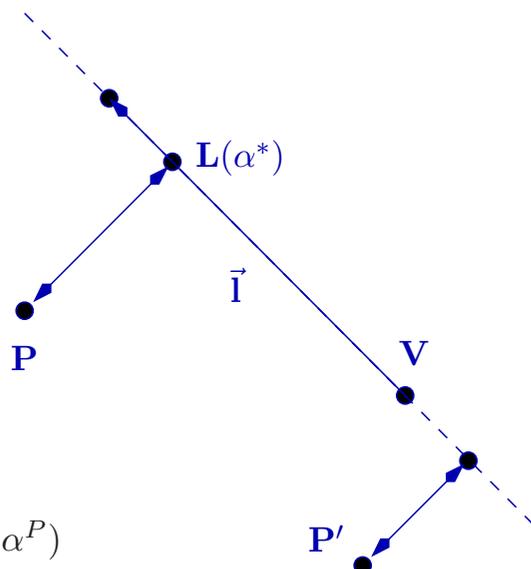
$$\mathcal{P}_L(\mathbf{P}) \in L \text{ mit } \|\mathcal{P}_L(\mathbf{P}) - \mathbf{P}\| = d_L(\mathbf{P})$$

1. Projektion von  $\mathbf{P}$  auf Gerade:

$$\mathbf{L}(\alpha^*) = \mathbf{V} + \alpha^* \vec{\mathbf{I}} \text{ für ein } \alpha^* \in \mathbb{R}.$$

Es muss gelten:

$$\begin{aligned} & ((\mathbf{P} - \mathbf{L}(\alpha^*)) \cdot \vec{\mathbf{I}}) = 0 \\ \Leftrightarrow & ((\mathbf{P} - (\mathbf{V} + \alpha^* \vec{\mathbf{I}})) \cdot \vec{\mathbf{I}}) = 0 \\ \Leftrightarrow & \alpha^* = \frac{((\mathbf{P} - \mathbf{V}) \cdot \vec{\mathbf{I}})}{(\vec{\mathbf{I}} \cdot \vec{\mathbf{I}})} \end{aligned}$$



2. Mit  $\alpha^P = \text{clip}(\alpha^*, [0, 1])$  ist  $\mathcal{P}_L(\mathbf{P}) = \mathbf{L}(\alpha^P)$



## 6.1.2 Lozenges (Rhombus)



**Ansatz:** Analog wie Kapsel, nur *Rechteck*  $R$  als Basisgeometrie:

$$R = \{\mathbf{V} + \alpha_1 \vec{e}_1 + \alpha_2 \vec{e}_2 : \alpha_1, \alpha_2 \in [0, 1]\}, \quad \text{mit } \vec{e}_1 \perp \vec{e}_2$$

**Projektion**  $\mathcal{P}_R(\mathbf{P})$  von  $\mathbf{P}$  auf  $R$ :

1. Projiziere  $\mathbf{P}$  auf Ebene

$$\mathbf{V}(\alpha_1^*, \alpha_2^*) = \mathbf{V} + \alpha_1^* \vec{e}_1 + \alpha_2^* \vec{e}_2, \\ (\alpha_1^*, \alpha_2^*) \in \mathbb{R}^2:$$

$$(\vec{e}_i \cdot (\mathbf{P} - \mathbf{V}(\alpha_1^*, \alpha_2^*))) = 0, \quad i = 1, 2$$

$$\Leftrightarrow (\vec{e}_i \cdot (\mathbf{P} - (\mathbf{V} + \alpha_1^* \vec{e}_1 + \alpha_2^* \vec{e}_2))) = 0, \quad i = 1, 2$$

$$(\vec{e}_1 \cdot \vec{e}_2) = 0 \quad \Leftrightarrow \quad \begin{pmatrix} \alpha_1^* \\ \alpha_2^* \end{pmatrix} = \begin{pmatrix} \frac{(\vec{e}_1 \cdot (\mathbf{P} - \mathbf{V}))}{(\vec{e}_1 \cdot \vec{e}_1)} \\ \frac{(\vec{e}_2 \cdot (\mathbf{P} - \mathbf{V}))}{(\vec{e}_2 \cdot \vec{e}_2)} \end{pmatrix}$$

2.  $\mathcal{P}_R(\mathbf{P})$  durch Clipping:

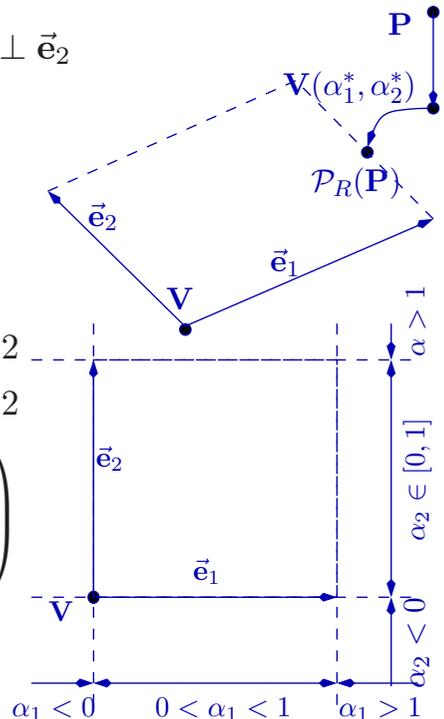


$$\mathcal{P}_R(\mathbf{P}) = \mathbf{V}(\alpha_1^P, \alpha_2^P) \quad \text{mit } \alpha_i^P = \text{clip}(\alpha_i^*, [0, 1])$$

Prof. Dr. Andreas Kolb  
Computer Graphics & Multimedia Systems

-Folie 6-7-

Computergraphik II



## 6.1.3 Verknüpfung einfacher impliziter Geometrien



**Ansatz:** Minimaler Abstand

**Vorgabe:** Basisgeometrie aus mehreren Teilen  $B = \bigcup \{B_1, \dots, B_k\}$ ,  
Schwellwert  $s$

**Abstand:** Der Abstand ergibt sich zu:  $d_B(\mathbf{P}) = \min\{d_{B_i}(\mathbf{P}) : i = 1, \dots, k\}$

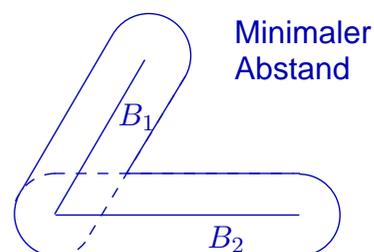
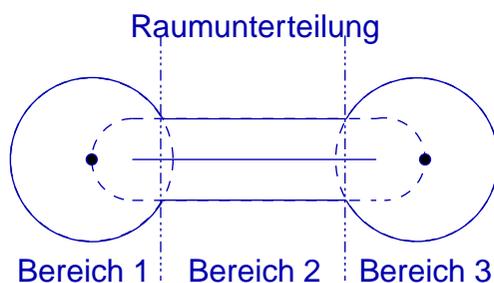
**Ansatz:** Raumunterteilung

**Vorgabe:** 1. Basisgeometrie besteht aus mehreren Teilen  $B_1, \dots, B_k$

2. Schwellwerte  $s_1, \dots, s_k$

3. Disjunkte Zerlegung des Raumes:  $\mathbb{R}^3 = \bigcup_{i=1}^k U_i$

**Abstand:** Der Abstand ergibt sich zu:  $d(\mathbf{P}) = d_{B_i}(\mathbf{P}) \Leftrightarrow \mathbf{P} \in U_i$



Prof. Dr. Andreas Kolb  
Computer Graphics & Multimedia Systems

-Folie 6-8-

Computergraphik II

## 6.2 Verallgemeinerte Implizite Funktionen

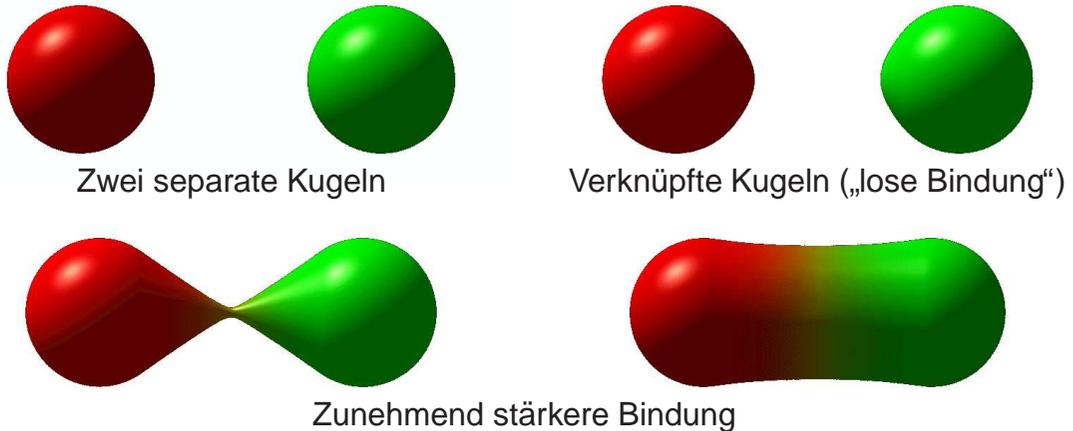


**Bislang:** Einfache implizite Geometrien, Komposition ergibt stets scharfe Kanten

**Ziel:** Komposition komplexer glatter Modelle, evtl. mehr Freiheitsgrade

**Idee:** Eingeschränkter und kontrollierbarer Einflussbereich der einzelnen Basisgeometrien

**Beispiele:**



## 6.2 Verallgemeinerte Implizite Funktionen ...



### Definition: Feldfunktion und Potentialfunktion

**Ansatz:** Verwende zusätzlich zur Distanzfunktion eine *Feldfunktion*

$$f : \mathbb{R} \rightarrow \mathbb{R}:$$

$$F(\mathbf{P}) = f(d_B(\mathbf{P})) \quad (\text{Potentialfunktion})$$

**Verknüpfung:** Mehrere Potentialfunktionen  $F_i$  mit Basisgeometrien  $B_i$  werden einfach addiert

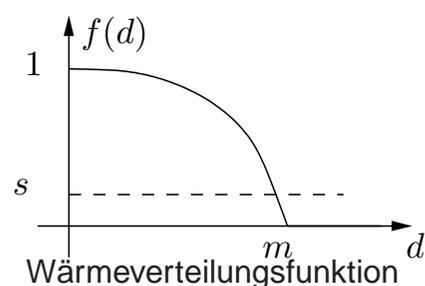
$$F(\mathbf{P}) = \sum_i F_i(\mathbf{P}) = \sum_i f_i(d_{B_i}(\mathbf{P}))$$

**Implizite Fläche:** Definiert sich als

$$M_s = \{\mathbf{P} : F(\mathbf{P}) = s\} \quad \text{für Schwellwert } s$$

**Beispiel:** Wärmeverteilungsfunktion in  
Abhängigkeit vom Abstand  $d$ :

$$f(d) = \max\left\{1 - \frac{d^2}{m^2}, 0\right\}$$

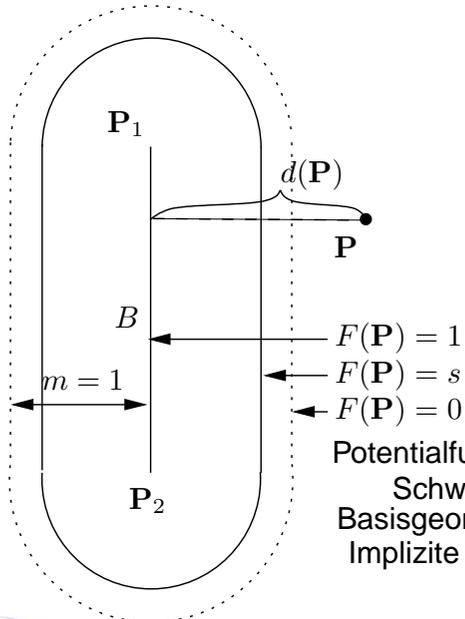


$m$  max. Distanz zur Basisgeometrie.



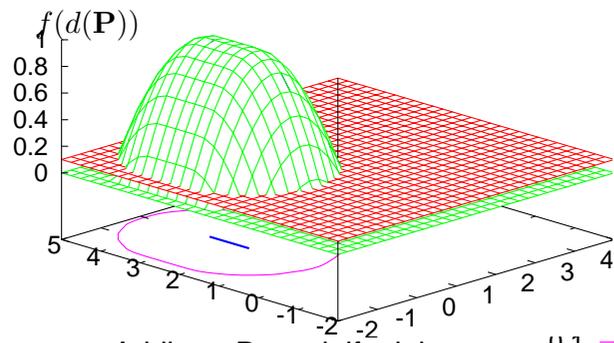
### Beispiel: Verknüpfung von Zylindern

Potentialfunktion: Kurven mit konstanten Schwellwerten:



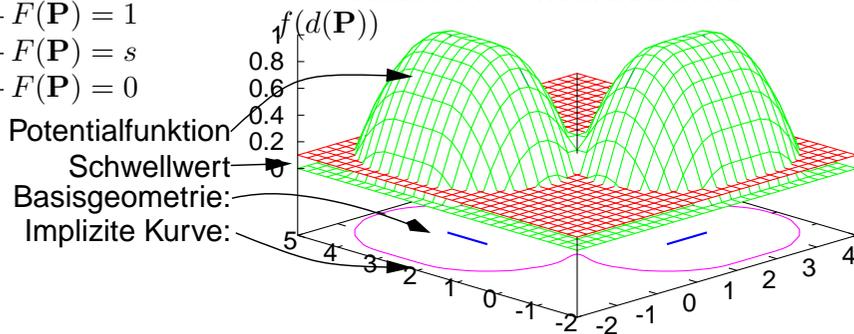
Eine Potentialfunktion:

0.1



Addierte Potentialfunktionen:

0.1

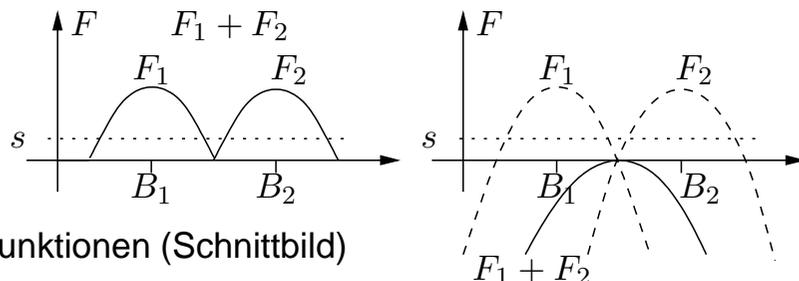


## 6.2 Verallgemeinerte Implizite Funktionen ...



### Anforderungen an die Feldfunktion $f$

**Positivität:** Die Bedingung  $f \geq 0$  ist für implizite Flächen sehr wichtig:



Addierte Potentialfunktionen (Schnittbild)

**Lokalität:** Träger  $\text{supp}(f)$  legt Einflussbereich fest

**Stetigkeitsgrad der Geometrie** durch Stetigkeit der Feldfunktion bestimmt

**Beispiele:** 1. Quadr. Wärmeverteilung:  $f(d) = \left(\max\left(1 - \frac{d^2}{m^2}, 0\right)\right)^2$ ,  $C^1$ -stetig

2. Kubische Wärmeverteilung:  $f(d) = \left(\max\left(1 - \frac{d^2}{m^2}, 0\right)\right)^3$ ,  $C^2$ -stetig

3. Exponentialfunktion:  $f(d) = e^{-\frac{d^2}{m^2}}$  (kein lokaler Träger!)

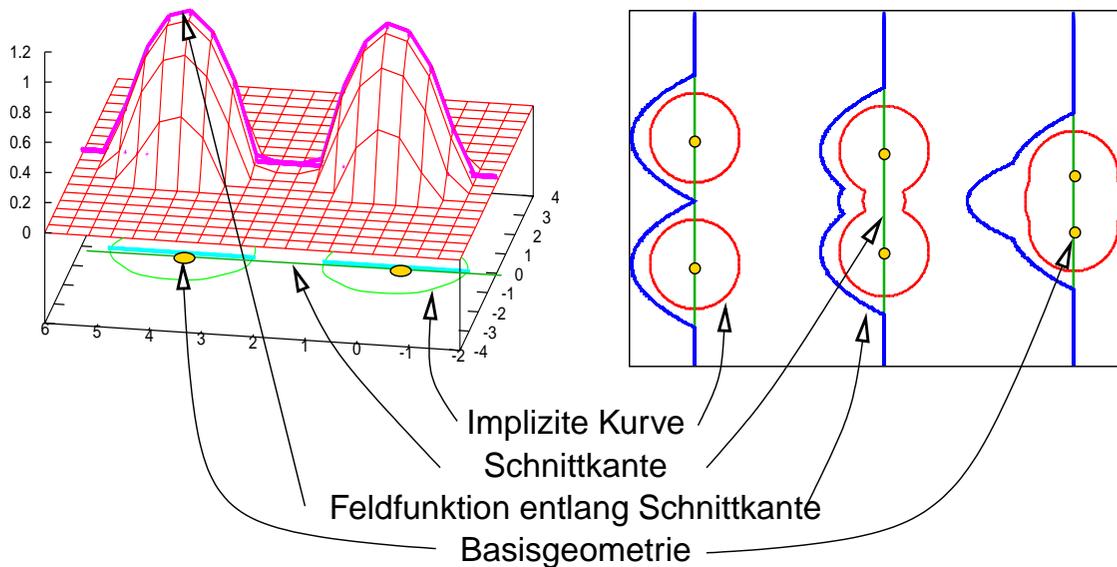


## 6.2 Verallgemeinerte Implizite Funktionen ...



### Beispiel verschiedener Feldfunktionen

Schnittbild der Wärmeverteilungsfunktion:



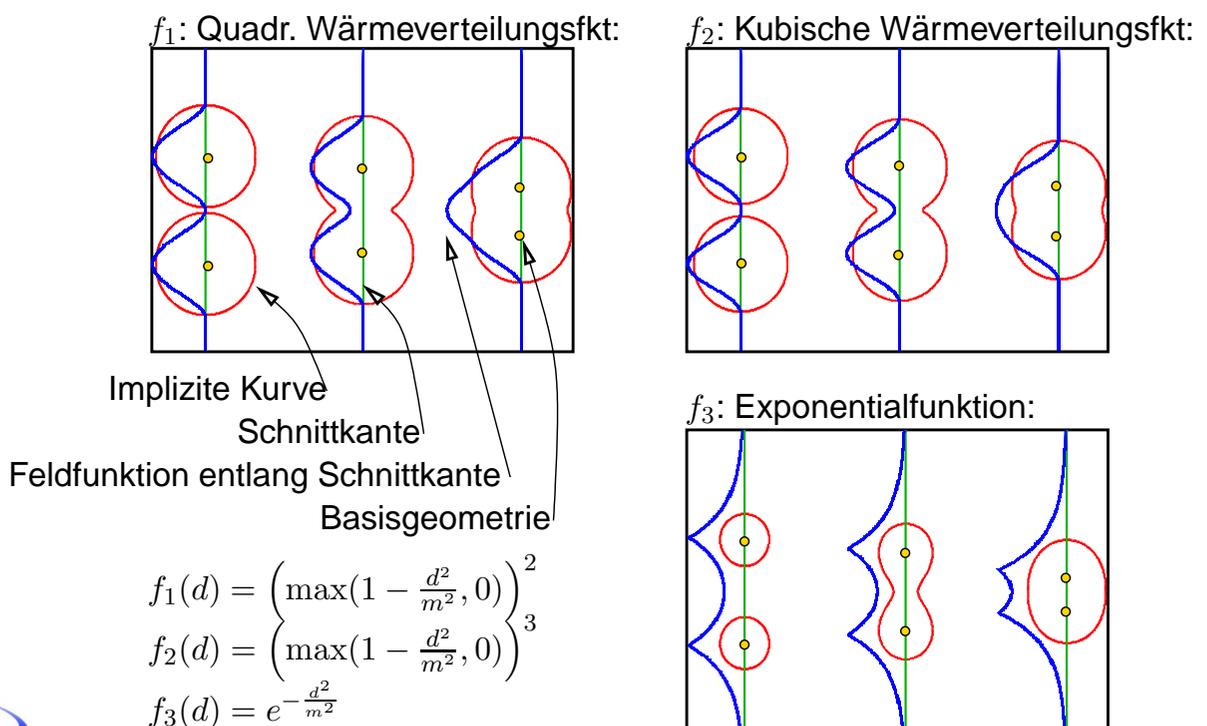
Beachte:  $C^0$ -Stetigkeit am Übergang zu 0 führt zu „Spitzen“



## 6.2 Verallgemeinerte Implizite Funktionen ...



### Beispiel verschiedener Feldfunktionen (Forts.)



## 6.3 Polygonalisierung impliziter Flächen



**Ziel:** Polygonale Approximation z.B. für Echtzeitgraphik-Anwendungen

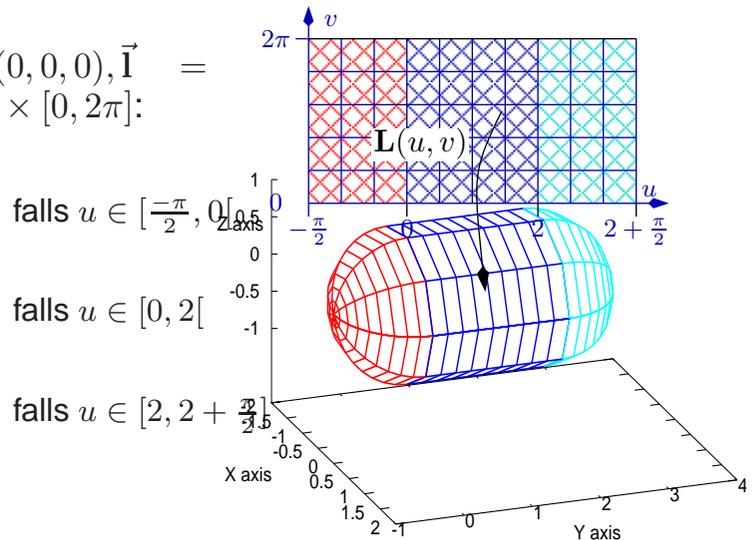
**Ansatz:** Polygonalisierung durch Parametrisierung

**Idee:** Suche parametrische Darstellung & zerlege Parametergebiet

**Problem:** Nur für einfache implizite Geometrien möglich

**Beispiel:** Kugel mit  $\mathbf{V} = (0, 0, 0), \vec{\mathbf{1}} = (0, 2, 0), s = 1$  über  $[-\frac{\pi}{2}, 2 + \frac{\pi}{2}] \times [0, 2\pi]$ :

$$\mathbf{L}(u, v) = \begin{cases} \begin{pmatrix} \cos(u) \cos(v) \\ \cos(u) \sin(v) \\ \sin(u) \end{pmatrix} & \text{falls } u \in [-\frac{\pi}{2}, 0[ \\ \begin{pmatrix} \cos(v) \\ \sin(v) \\ u \end{pmatrix} & \text{falls } u \in [0, 2[ \\ \begin{pmatrix} \cos(u-2) \cos(v) \\ \cos(u-2) \sin(v) \\ 2 + \sin(u-2) \end{pmatrix} & \text{falls } u \in [2, 2 + \frac{\pi}{2}] \end{cases}$$



## 6.3 Polygonalisierung impliziter Flächen ...



### Beliebige implizite Geometrien (2D-Marching Cube Algorithmus)

**Ausgangslage:** Implizite Geometrie

$$\{\mathbf{P} : F(\mathbf{P}) = s\} = \{\mathbf{P} : h(\mathbf{P}) = F(\mathbf{P}) - s = 0\}$$

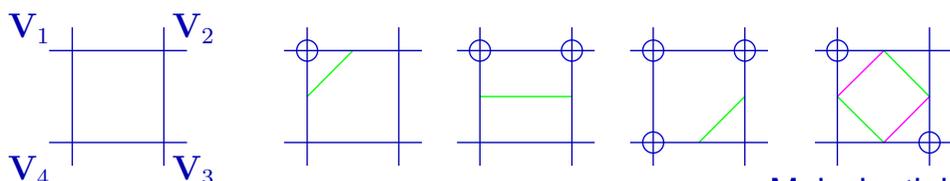
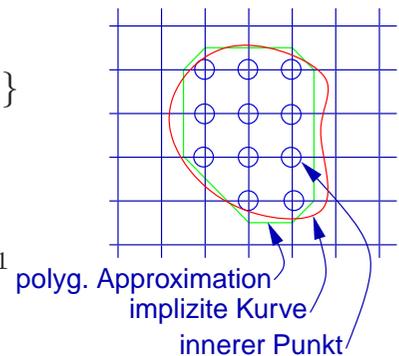
**Diskretisierung:** Regelmäßiges Gitter:

$$\mathbf{P}_{ij} = \mathcal{O} + \begin{pmatrix} i\Delta_x \\ j\Delta_y \end{pmatrix}, \quad i \in \{0, \dots, N_x\}, j \in \{0, \dots, N_y\}$$

**Klassifikation:** Vier grundsätzliche Fälle:

$$\mathbf{V}_1 = \mathbf{P}_{ij}, \mathbf{V}_2 = \mathbf{P}_{i+1,j}, \mathbf{V}_3 = \mathbf{P}_{i+1,j+1}, \mathbf{V}_4 = \mathbf{P}_{i,j+1}$$

$$\mathbf{V}_k \text{ Innen/Außen} \Leftrightarrow h(\mathbf{V}_k) < 0 \text{ bzw. } > 0$$



Mehrdeutigkeit!



## 6.3 Polygonalisierung impliziter Flächen ...



### Beliebige implizite Geometrien (2D-Marching Cube Algorithmus) (Forts.)

Schnittpunkt-Kriterium für Schnitt Kante  $\overline{\mathbf{V}_k \mathbf{V}_l}$  schneidet implizite Kurve

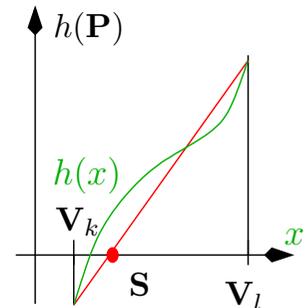
$$S \Leftrightarrow h(\mathbf{V}_k) \cdot h(\mathbf{V}_l) < 0$$

Schnittpunktabschätzung mittels Regula-Falsi (lineare Approximation von  $h$ ):

$$\text{Schnittpunkt } S = \overline{\mathbf{V}_k \mathbf{V}_l} \cap \{\mathbf{P} : h(\mathbf{P}) = 0\}$$

$$\text{Approximation : } S \approx (1 - \alpha)\mathbf{V}_k + \alpha\mathbf{V}_l$$

$$\text{mit } \alpha = h(\mathbf{V}_k) / (h(\mathbf{V}_k) - h(\mathbf{V}_l))$$



Herleitung (Strahlensatz):

$$\frac{h(\mathbf{V}_l)}{-h(\mathbf{V}_k)} = \frac{1 - \alpha}{\alpha} \Leftrightarrow \alpha \cdot h(\mathbf{V}_l) = \alpha \cdot h(\mathbf{V}_k) - h(\mathbf{V}_k) \Leftrightarrow \alpha = \frac{h(\mathbf{V}_k)}{h(\mathbf{V}_k) - h(\mathbf{V}_l)}$$



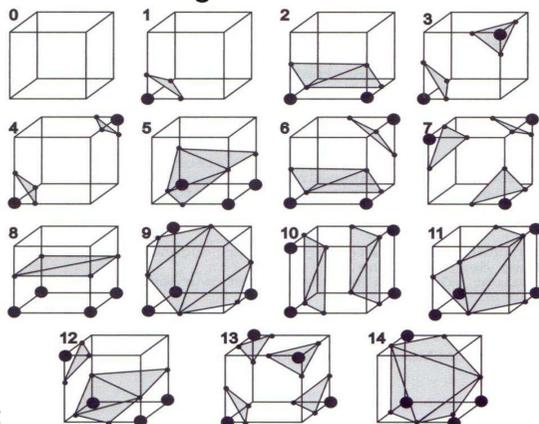
## 6.3 Polygonalisierung impliziter Flächen ...



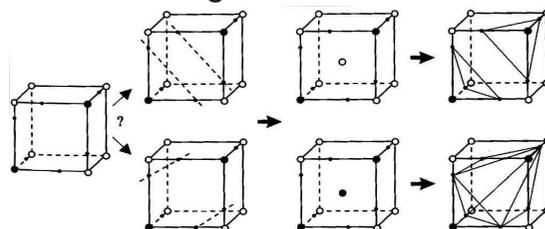
### Marching Cube Algorithmus (3D)

- grundsätzlich  $2^8 = 256$  mögliche Kombinationen; 15 relevante Fälle
- eine größere Zahl von Problemfällen: fünf Stück
- Linearisierung: Schnittpunkte werden zu Dreiecken verknüpft (Lookup-Table)

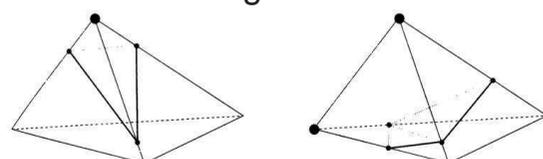
#### 3D-Marching Cube: Klassifikation



#### Behandlung eines Problemfalls



#### Marching Tetrahedra



## 6.3 Polygonalisierung impliziter Flächen ...



### Algorithmus: Marching Cubes (3D)

**Lookup-Tabelle (LUT):** Speichert für alle  $2^8 = 256$  Fälle

- Anzahl Schnittpunkte (LUT[k].N) & zugehörigen Kanten (LUT[k].S[j].V1, LUT[k].S[j].V2)
- Anzahl der Dreiecke (LUT[k].M) & Verbindung der Punkte zu Dreiecken (LUT[k].T[j].S1, LUT[k].T[j].S2, LUT[k].T[j].S3)

**Algorithmus** zur Erzeugung der Polygonoberfläche

```
for ( voxel ∈ grid ) {  
  compute LUT-index k from  $\text{sgn}(h(\mathbf{V}_i))$ ,  $i = 1, \dots, 8$   
  for ( i = 0; i < LUT[k].N; i ++ ) {  
    compute intersection  $S_i$  along LUT[k].S[i].V1, LUT[k].S[i].V2  
    add  $S_i$  to vertex list  
  }  
  for ( i = 0; i < LUT[k].M; i ++ )  
    add triangle LUT[k].T[i].S1, LUT[k].T[i].S2, LUT[k].T[i].S3  
    to triangle list  
}
```



## 6.3 Polygonalisierung impliziter Flächen ...



### Normalen einer impliziten Fläche

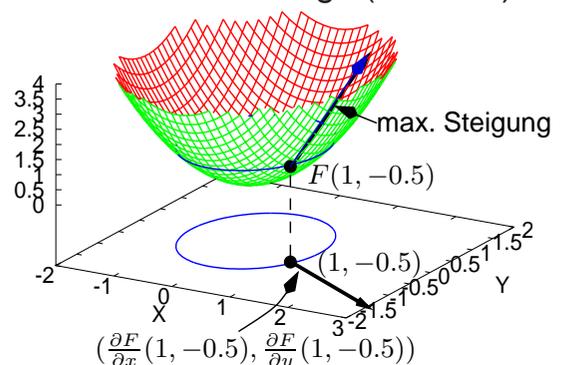
**Ziel:** Effiziente Bestimmung von Normalenvektoren für implizite Fläche

**Beobachtung:** Ableitung „zeigt“ in Richtung des steilsten Anstiegs (s. unten)

Beispiel: Funktion  $F(x, y) = x^2 + y^2$

$$\left( \frac{\partial F}{\partial x}(x, y), \frac{\partial F}{\partial y}(x, y) \right) = (2x, 2y)$$

$$\text{Gradient: } \nabla F = \text{grad}[F] := \left( \frac{\partial F}{\partial x}, \frac{\partial F}{\partial y} \right)$$



**Implizite Fläche:** Funktion  $F : \mathbb{R}^3 \rightarrow \mathbb{R}$

- Definitionsbereich = 3D  $\Rightarrow$  Gradient Vektor in 3D:  $\nabla F = \left( \frac{\partial F}{\partial x}, \frac{\partial F}{\partial y}, \frac{\partial F}{\partial z} \right)$
- Implizite Fläche = Iso-Fläche (konstanter Funktionswert)
- Richtung max. Steigung und Fläche konstanten Wertes sind orthogonal  $\Rightarrow$  Gradient orthogonal zur Fläche



## 6.3 Polygonalisierung impliziter Flächen ...



Frage: Warum zeigt  $\nabla F$  in Richtung des max. Abstiegs?

**Gegeben:** ○ Funktion  $F : \mathbb{R}^3 \rightarrow \mathbb{R}$  (z.B. Temperatur),  $\nabla F = \left( \frac{\partial F}{\partial x}, \frac{\partial F}{\partial y}, \frac{\partial F}{\partial z} \right)^T$

○ Kurve  $\mathbf{X}(u) = (x(u), y(u), z(u))^T$  (Bewegungsbahn)

**Betrachte**  $F$  entlang  $\mathbf{X}$ , also  $F(x(u), y(u), z(u))$  und dessen Veränderung  $\frac{dF(\mathbf{X}(u))}{du}$

$$\text{Kettenregel: } \frac{dF(\mathbf{X}(u))}{du} = \frac{\partial F}{\partial x} \cdot \frac{dx(u)}{du} + \frac{\partial F}{\partial y} \cdot \frac{dy(u)}{du} + \frac{\partial F}{\partial z} \cdot \frac{dz(u)}{du} = (\nabla F \cdot \mathbf{X}'(u))$$

$\mathbf{X}'(u)$  entspricht der Geschwindigkeit entlang der Bahn  $\mathbf{X}$

**Veränderung wird maximal** bei konstanter Geschwindigkeit  $\|\mathbf{x}'(u)\|$ , wenn  $\nabla F \parallel \mathbf{x}'(u)$

**Veränderung ist null**, d.h.  $\mathbf{X}$  läuft auf einer Isofläche von  $F$ , wenn  $\nabla F \perp \mathbf{x}'(u)$

