



# Level Set Methoden I

Seminar Bildsegmentierung und Computer Vision

Sarah Eve Wiedemann

7.November 2005

# Level Set Methoden

von Osher und Sethian

Grundidee:

Hamilton Jacobi Ansatz zur numerischen Lösung von zeitabhängigen Gleichungen für implizit sich bewegende Oberflächen

# Gliederung

- Bewegungen in extern erzeugten Geschwindigkeitsfeldern
  - Hilfreiche Methoden
    - Konvektion
    - Upwind Differencing
    - Hamilton-Jacobi Essentially Nonoscillatory (ENO) Polynomial Interpolation
    - Hamilton-Jacobi Gewichtete ENO (WENO)
    - TVD (Total Variation Diminishing) Runge Kutta Methode
  
- Bewegungen in intern erzeugten Geschwindigkeitsfeldern
  - Gleichungen
    - Bewegungsgleichung
    - Wärmegleichung
  - Numerische Diskretisierung
  
- Hamilton-Jacobi Gleichungen
  - Erhaltungsgesetz
  - Eulersche Vorwärtszeitdiskretisierung

# Konvektion

## Grundlegende Funktionen:

- Geschwindigkeit auf einer Oberfläche  $V(\vec{x}) = \langle u, v, w \rangle, \vec{x} = (x, y, z)$
- Implizite Funktion, die 0 wird für jeden Punkt  $\vec{x}$  an der Kontur  $\Phi(\vec{x})$

Alle Punkte einer Oberfläche bewegen sich mit der gleichen Geschwindigkeit:

$$d\vec{x} / dt = \vec{V}(\vec{x}) \quad \text{für alle } \vec{x} \text{ mit } \Phi(\vec{x}) = 0$$

*„Lagrange Formulierung der Oberflächenevolutionsgleichung“*

## Problem:

Es gibt eine unendliche Anzahl an Punkten (Dimension 1 ausgenommen)

## Lösung:

Front in unendliche Anzahl von Stücken diskretisieren

**Aber:** - Verzerrung des Geschwindigkeitsfeldes

- Fehlende Genauigkeit, falls keine häufige Änderung der Diskretisierung

# Konvektion (II)

## ■ Probleme:

- Instabilität
- Deformationen der Oberflächenelemente
- komplizierte Prozeduren für das „topologische“ Reparieren von Grenzflächen nötig

## ■ Besserer Ansatz:

- implizite Funktion  $\Phi$

$$\Phi_t + \vec{V} \nabla \Phi = 0 \quad (1)$$

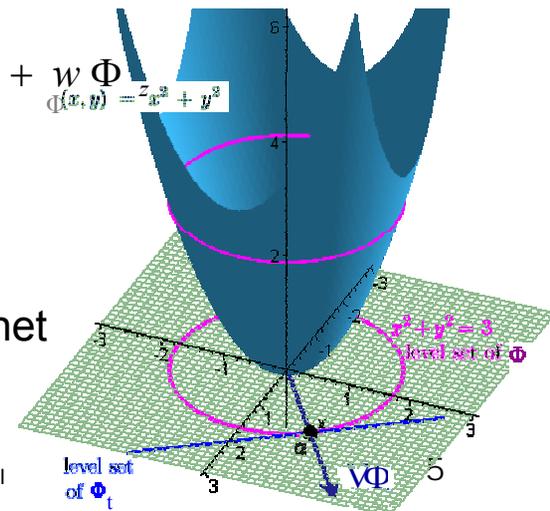
mit:  $\Phi_t$  partielle Ableitung nach der Zeit

$$\nabla \Phi = (\Phi_x, \Phi_y, \Phi_z)$$

$$\text{Gradient mit } \vec{V} * \nabla \Phi = u \Phi_x + v \Phi_y + w \Phi_z$$

Diese Funktion

- Beschreibt die Bewegung der gewünschten Grenzfläche
- Ist die **Level Set Gleichung**
- Wird auch als **einfache Konvektionsgleichung** bezeichnet



Quelle:  
[http://www.ualberta.ca/dept/math/gauss/fcm/calculus/multvrb/basic/ImplctFnctns/implct\\_fnctn\\_illstrn1.htm](http://www.ualberta.ca/dept/math/gauss/fcm/calculus/multvrb/basic/ImplctFnctns/implct_fnctn_illstrn1.htm)

# Konvektion (III)

## Probleme der Level Set Gleichung

- (1) Realisierung im kartesischen Koordinatensystem kompliziert, falls Geschwindigkeitsfeld nur auf der Grenzfläche definiert ist
- (2) Grenzfläche vermittelt den Eindruck stillzustehen

## Lösungsansätze

- (1) Die Level Set Gleichung wird so geschrieben, dass  $\vec{V}$  auch dort wo  $\Phi(\vec{x}) \neq 0$  definiert ist.  
In der Numerik ist es oft ausreichend, dass  $\vec{V}$  auf einem Band definiert ist, dass die Grenzfläche enthält.
- (2)  $\vec{V}$  stetig nahe der Oberfläche (aber:  $\vec{V}$  auf eine Menge von stetigen Funktionen zu beschränken, löst das Problem nicht.)  
Ein Band  $\varepsilon$  mit  $\Delta x \ll \varepsilon$  wird um  $\vec{V}$  gelegt.  $\varepsilon$  muss dabei relativ groß sein, um die Ableitung des Geschwindigkeitsfeldes zu minimieren.

## Konvektion (IV)

Die Grenzflächenbewegung wird bestimmt durch die Ableitung der zur Grenzfläche tangentialen Geschwindigkeit. Die Ableitung der Normale ist bedeutungslos.

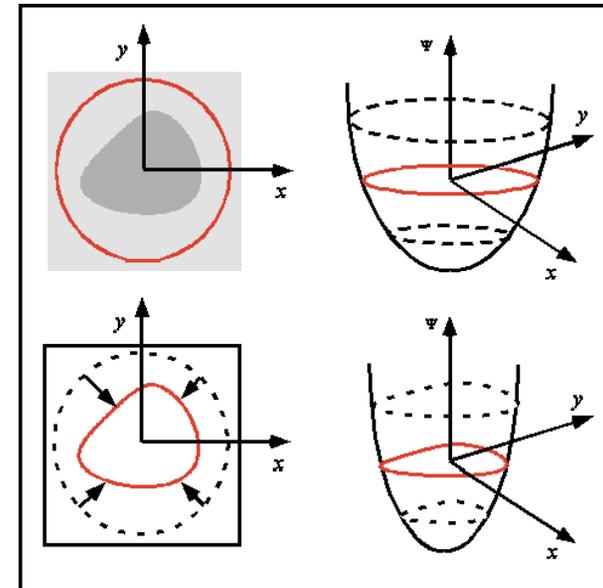
Bestimmung des Punktes, an dem die Geschwindigkeitsableitung minimal ist:

Die Grenzfeldgeschwindigkeit wird auf eine Konstante in Normalenrichtung der Grenzfläche beschränkt, so dass gilt:

- $\vec{V}(\vec{x}) = \vec{V}(\vec{x}_c)$
- $\vec{x}_c$  nächster Punkt der Grenzflächen zu  $\vec{x}$
- Geschwindigkeit in Normalenrichtung konstant.

### Bemerkung:

- Das Geschwindigkeitsfeld kann von mehreren externen Quellen bestimmt werden.
- Die Grenzflächengeschwindigkeit hängt von Raum **und** Zeit ab, oft wird die Zeit aber wegen der Kürze bei der Notation weggelassen.



# Upwind Differencing

Auswertung von  $\Phi$  mit Hilfe numerischer Methoden:

- $\Phi$  und  $\vec{V}$  sind in jedem Gitterpunkt eines kartesischen Koordinatensystems definiert

es gilt:  $\Phi^n = \Phi(t^n)$   
 $\Phi^{n+1} = \Phi(t^{n+1})$  mit  $t^{n+1} = t^n + \Delta t$

## Einfache Methode 1.Ordnung für die Zeitdiskretisierung der Level Set Gleichung

$$\left( (\Phi^{n+1} - \Phi^n) / \Delta t \right) + \vec{V}^n \nabla \Phi^n = 0 \quad (2)$$

mit:  $\vec{V}^n$  Externes Geschwindigkeitsfeld zu Zeit  $t_n$   
 $\nabla \Phi^n$  Gradientoperator mit Werten  $\Phi$  zu Zeit  $t_n$

## Eulersche Vorwärtsgleichung

## Upwind Differencing (II)

Aus Gleichung (2) ergibt sich:

$$((\Phi^{n+1} - \Phi^n) / \Delta t) + u^n \Phi_x^n + v^n \Phi_y^n + w^n \Phi_z^n = 0$$

Reduzierung auf  $u^n \Phi_x^n$

- Lösung für  $v^n \Phi_y^n$  und  $w^n \Phi_z^n$  analog mit dimension-by-dimension Technik

$$\Rightarrow ((\Phi^{n+1} - \Phi^n) / \Delta t) + u^n \Phi_x^n = 0$$

Die Variable  $u^n$

- zeigt an, ob die Werte von  $\Phi$  sich nach rechts oder links bewegen.
- kann räumlich variieren
  - daher Betrachtung eines festen Gitterpunktes  $x_i$

$$\Rightarrow ((\Phi_i^{n+1} - \Phi_i^n) / \Delta t) + u_i^n (\Phi_x)_i^n = 0$$

mit  $(\Phi_x)_i$  räumliche Ableitung von  $\Phi(x_i) = \Phi_i$

## Upwind Differencing (III)

### Fallunterscheidung

- $u_i > 0$   $\Phi$  bewegt sich von links nach rechts. Die Charakteristikenmethode zeigt auf, dass man nach links ausgehend von  $x_i$  schauen muss, um den Wert von  $\Phi$  im letzten Zeitschritt zu bestimmen.
- $u_i < 0$   $\Phi$  bewegt sich von rechts nach links. Die Charakteristikenmethode zeigt auf, dass man nach rechts sehen sollte, um den Wert von  $\Phi_i$  zum Zeitpunkt  $t^{n+1}$  zu bestimmen.
- $u_i = 0$   $u_i \bullet (\Phi_x)_i = 0 \Rightarrow$  Keine Approximation

$\Rightarrow D^-\Phi (= \Phi_x^-)$  soll  $\Phi_x$  approximieren, wenn  $u_i > 0$

$D^+\Phi (= \Phi_x^+)$  soll  $\Phi_x$  approximieren, wenn  $u_i < 0$

Diese Methode eine Approximation zu wählen, nennt sich

**Upwind differencing.**

## Hilfsmittel

Die Kombination der Eulerschen Vorwärts-Zeitdiskretisierung mit dem upwind differencing Schema ist eine stetige finite Differenzenapproximation für die Level Set Gleichung, wenn der Approximationsfehler gegen 0 konvergiert.

### ■ *Lax-Richtmyer Equivalence Theorem*

Ein stetiges finites Differenzenschema einer PDE, die ein gut gestelltes Anfangswertproblem hat, ist genau dann konvergent, wenn sie stabil ist.

### ■ *Courant-Friedrichs-Lewy Kondition (CFL Kondition)*

Numerische Wellen sollen sich mindestens genauso schnell wie physikalische verbreiten.

$$\Delta x / \Delta t \geq |u|$$

*CFL Zeitschrittbegrenzung*

$$\Delta t (\max \{|u|\} / \Delta x) = \alpha \quad \text{mit } 0 < \alpha < 1$$

- $\max\{|u|\}$  größter Wert von  $|u|$  auf dem ganzen Gitter
- ! In der Realität genügt der größte Wert von  $|u|$  auf der Grenzfläche.

Alternative zum upwind differencing ist das *central differencing*.

# Hamilton-Jacobi ENO

## ■ Idee

Essentially nonoscillatory (ENO) polynomial Interpolation für die numerische Lösung des Erhaltungsgesetzes

- Die Hamilton-Jacobi ENO (HJ ENO) Methode erlaubt eine Erweiterung des eindimensionalen upwind differencing zu höherdimensionaler, räumlicher Genauigkeit durch bessere numerische Approximation von  $\Phi_x^+$  und  $\Phi_x^-$ .

## ■ Grundvorgehen

Verwendung des glattestmöglichen Interpolationspolynom, um  $\Phi$  zu bestimmen.

Differentiation dieses Polynoms um  $\Phi_x$  zu erhalten

# Erhaltungsgesetz

- Eindimensionales Erhaltungsgesetz

$$u_t + F(u)_x = 0$$

$u$  erhaltene Masse  
 $F(u)$  Flussfunktion

- Beispiele

- Kontinuitätsgleichung

$$\delta_t + (\delta w)_x = 0$$

$\delta$  Dichte des Materials  
 $w$  Geschwindigkeitsfeld des Fluides

- Burger Gleichung

$$u_t + (u^2 / 2)_x = 0$$

skalares Erhaltungsgesetz

## Hamilton Jacobi ENO (II)

- **Vorgehen**

Newton Interpolationspolynom  $D_i^0 \Phi = \Phi_i$  mit Netzpunkt  $i$ .

=> An jedem Netzpunkt  $i$  gibt es einen Differenzenquotienten

$$D_{(i-1)/2}^1 \Phi = (D^- \Phi)_i$$

$$D_{(i+1)/2}^1 \Phi = (D_{i+1}^0 \Phi - D_i^0 \Phi) / \Delta x = (D^+ \Phi)_i$$

$$\Rightarrow D_i^2 \Phi = (D_{(i+1)/2}^1 \Phi - D_{(i-1)/2}^1 \Phi) / (2\Delta x)$$

$$D_{(i+1)/2}^3 \Phi = (D_{i+1}^2 \Phi - D_i^2 \Phi) / (3\Delta x)$$

- Ein Polynom folgender Form kann konstruiert werden:

$$\Phi(x) = Q_0(x) + Q_1(x) + Q_2(x) + Q_3(x)$$

$$\Phi_x(x_i) = Q_1'(x_i) + Q_2'(x_i) + Q_3'(x_i)$$

- Letzteres hilft  $(\Phi_x^+)_i$  und  $(\Phi_x^-)_i$  zu definieren.

## Hamilton Jacobi ENO (III)

- Um  $\Phi_x^+$  zu ermitteln, wähle man  $k=i-1$ , für  $\Phi_x^-$   $k=i$ .

- Es gilt: 
$$Q_1(x) = (D_{(k+1)/2}^1 \Phi)(x - x_i)$$

$$Q_1'(x_i) = D_{(k+1)/2}^1 \Phi$$

Falls  $|D_k^2 \Phi| \leq |D_{k+1}^2 \Phi|$  setze  $c = D_k^2 \Phi$  und  $k^* = k - 1$   
 sonst  $c = D_{k+1}^2 \Phi$  und  $k^* = k$

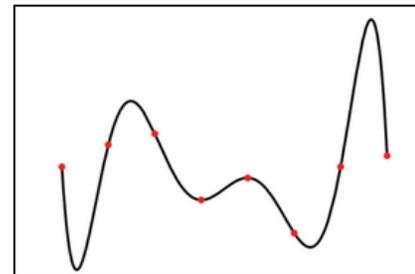
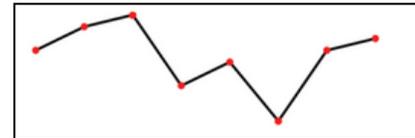
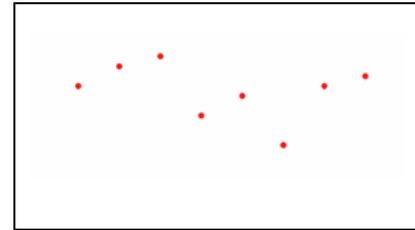
$$\Rightarrow Q_2(x) = c(x - x_k)(x - x_{k+1})$$

$$Q_2'(x_i) = c(2(i - k) - 1)\Delta x$$

Falls  $|D_{(k^*+1)/2}^3 \Phi| \leq |D_{(k^*+3)/2}^3 \Phi|$  setze  $c^* = D_{(k^*+1)/2}^3 \Phi$   
 sonst  $c^* = D_{(k^*+3)/2}^3 \Phi$

$$\Rightarrow Q_3(x) = c^* (x - x_{k^*})(x - x_{k^*+1})(x - x_{k^*+2})$$

$$Q_3'(x_i) = c^* (3(i - k^*)^2 - 6(i - k^*) + 2)(\Delta x)^2$$



Interpolation am grafischen  
Beispiel 15

## Hamilton Jacobi ENO (IV)

Bei der Berechnung von  $(\Phi_x^-)_i$  3. Ordnung eines HJ ENO Schemas haben wir eine Teilmenge  $\{\Phi_{i-3}, \Phi_{i-2}, \Phi_{i-1}, \Phi_i, \Phi_{i+1}, \Phi_{i+2}\}$ , die davon abhängt, welches Muster gewählt wird.

Es gibt 3 mögliche Approximationen von  $(\Phi_x^-)_i$ :

Es gilt:

$$\left. \begin{array}{l} v_1 = D^- \Phi_{i-2} \\ v_2 = D^- \Phi_{i-1} \\ v_3 = D^- \Phi_i \\ v_4 = D^- \Phi_{i+1} \\ v_5 = D^- \Phi_{i+2} \end{array} \right\} \begin{array}{l} \Phi_x^1 = (v_1 / 3) - (7v_2 / 6) + (11v_3 / 6) \\ \Phi_x^2 = (-v_2 / 6) + (5v_3 / 6) + (v_4 / 3) \\ \Phi_x^3 = (v_3 / 3) + (5v_4 / 6) - (v_5 / 6) \end{array}$$

### Ziel

Die Approximation mit dem geringsten Fehler zu wählen, indem man das glattestmögliche Interpolationspolynom von  $\Phi$  wählt.

## Gewichtete ENO Methode (WENO)

- Konvexe Kombination von 3 ENO Approximationen
- Unstetigkeiten bekommen möglichst wenig Gewicht

### HJ WENO

- Erweiterung der WENO Methode mit Hilfe des Hamilton Jacobi Regelwerks
- Löst die Level-Set Gleichung
- Fehler wird um Größenordnung 3 reduziert

$$\Phi_x = w_1 \Phi_x^1 + w_2 \Phi_x^2 + w_3 \Phi_x^3$$

$$0 \leq w_k \leq 1$$

$$w_1 + w_2 + w_3 = 1$$

- Optimale Wahl für Genauigkeit 5.Ordnung

$$w_1 = 0,1 + O((\Delta x)^2)$$

$$w_2 = 0,6 + O((\Delta x)^2)$$

$$w_3 = 0,3 + O((\Delta x)^2)$$

=>

$$0,1 \Phi_x^1 + 0,6 \Phi_x^2 + 0,3 \Phi_x^3$$

$$c_1(\Delta x)^2 \Phi_x^1 + c_2(\Delta x)^2 \Phi_x^2 + c_3(\Delta x)^2 \Phi_x^3$$

## Einschub: Hamilton-Jacobi Gleichungen

$$\Phi_t + H(\nabla\Phi) = 0$$

(6)

### Beispiel:

Konvektion in einem extern erzeugten Geschwindigkeitsfeld

$$H(\nabla\Phi) = \vec{V} \nabla\Phi$$

Level Set Gleichung

$$H(\nabla\Phi) = V_n |\nabla\Phi|$$

## Bestimmung der Gewichte

$$S_1 = 13/12(v_1 - 2v_2 + v_3)^2 + 1/4(v_1 - 4v_2 + 3v_3)^2$$

$$S_2 = 13/12(v_2 - 2v_3 + v_4)^2 + 1/4(v_2 - v_4)^2$$

$$S_3 = 13/12(v_3 - 2v_4 + v_5)^2 + 1/4(3v_3 - 4v_4 + v_5)^2$$

$$\begin{aligned} \Rightarrow \quad \alpha_1 &= 0,1/(S_1 + \varepsilon)^2 \\ \alpha_2 &= 0,6/(S_2 + \varepsilon)^2 \\ \alpha_3 &= 0,3/(S_3 + \varepsilon)^2 \\ \varepsilon &= 10^{-6} \max\{v_1^2, v_2^2, v_3^2, v_4^2, v_5^2\} + 10^{-99} \end{aligned}$$

Eine glatte Lösung hat wenig Variation und führt zu kleinem  $S_k$

$$\begin{aligned} \Rightarrow \quad \alpha_1 &\approx 0,1\varepsilon^2 \\ \alpha_2 &\approx 0,6\varepsilon^2 \\ \alpha_3 &\approx 0,3\varepsilon^2 \end{aligned}$$

## Bestimmung der Gewichte (II)

$$\Rightarrow w_1 = \alpha_1 / (\alpha_1 + \alpha_2 + \alpha_3)$$

$$w_2 = \alpha_2 / (\alpha_1 + \alpha_2 + \alpha_3)$$

$$w_3 = \alpha_3 / (\alpha_1 + \alpha_2 + \alpha_3)$$

$\Rightarrow$  approximative, optimale Gewichte, wenn die  $S_k$  klein genug sind und von  $\varepsilon$  dominiert werden:

$$w_1 = 0,1$$

$$w_2 = 0,6$$

$$w_3 = 0,3$$

Die Funktion  $(\Phi_x^+)$  kann analog mit folgender Teilmenge konstruiert werden:

$$\{\Phi_{i-2}, \Phi_{i-1}, \Phi_i, \Phi_{i+1}, \Phi_{i+2}, \Phi_{i+3}\}$$

## TVD Runge-Kutta Methode

### *HJ ENO/ HJ WENO*

- Raumgenauigkeit 5. Ordnung

### *Eulersche Vorwärts-Zeitdiskretisierung*

- Zeitdiskretisierung der Level Set Gleichung 1.Ordnung in Bezug auf die Zeit
- temporäre Rundungsfehler produzieren weniger signifikante Entartung der numerischen Lösung

- **Problem:**

Es gibt Fälle, wo eine Diskretisierung höherer Ordnung nötig ist, um eine akkurate numerische Lösung zu erhalten.

- **Lösung:**

Reduzierung der Totalvariation (Total variation diminishing (TVD)) Runge-Kutta (RK) Methode, um die Genauigkeit mit Hilfe des Ansatzes der Linienmethode in Bezug auf die Zeitdiskretisierung zu verbessern.

## TVD Runge-Kutta Methode (II)

- TVD RK Schema 1.Ordnung = Eulersche Vorwärtsmethode
- TVD RK Schema 2.Ordnung = Standard RK Schema 2.Ordnung

### Vorgehen:

- Euler Schritt, um die Lösung von  $t^n$  auf  $t^n + \Delta t$  anzunähern

$$((\Phi^{n+1} - \Phi^n) / \Delta t) + \vec{V}^n \nabla \Phi^n = 0$$

- 2. Eulerschritt

$$((\Phi^{n+2} - \Phi^{n+1}) / \Delta t) + \vec{V}^{n+1} \nabla \Phi^{n+1} = 0$$

- Mittelungsschritt

$$\Phi^{n+1} = \frac{1}{2} \Phi^n + \frac{1}{2} \Phi^{n+2}$$

# Gliederung

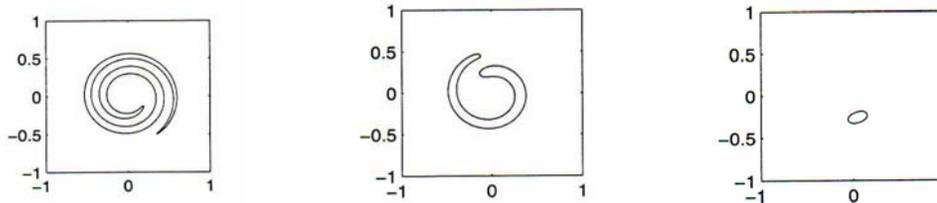
- Bewegungen in extern erzeugten Geschwindigkeitsfeldern
  - Hilfreiche Methoden
    - Konvektion
    - Upwind Differencing
    - Hamilton-Jacobi Essentially Nonoscillatory (ENO) Polynomial Interpolation
    - Hamilton-Jacobi Gewichtete ENO (WENO)
    - TVD (Total Variation Diminishing) Runge Kutta Methode
  
- Bewegungen in intern erzeugten Geschwindigkeitsfeldern
  - Gleichungen
    - Bewegungsgleichung
    - Wärmeleichung
  - Numerische Diskretisierung
  
- Hamilton-Jacobi Gleichungen
  - Erhaltungsgesetz
  - Eulersche Vorwärtszeitdiskretisierung

## Motion Involving Mean Curvature- Gleichung der Bewegung

- Bis jetzt Betrachtung extern erzeugter Geschwindigkeitsfelder  $\vec{V}(\vec{x}, \vec{t})$
- Im folgenden Betrachtung selbst-generierter Geschwindigkeitsfelder  $\vec{V}(\vec{x}, \vec{t})$ , die direkt von der Level Set Funktion  $\Phi$  abhängen

$$\vec{V} = -b \kappa \vec{N}$$

- $b < 0$  Grenzfläche bewegt sich in konvexe Richtung, so dass der Kreis wächst statt zu schrumpfen
- $b > 0$  Grenzfläche bewegt sich konkav, so dass der Kreis zu einem Punkt zusammenschrumpft und verschwindet



- Spezifikation der Tangentialkomponente generell nicht nötig für ein Geschwindigkeitsfeld – **Level Set Gleichung**

$$\Phi_t + V_n |\nabla \Phi| = 0$$

(3)

mit  $V_n$  Normalengeschwindigkeit

## Motion Involving Mean Curvature- Gleichung der Bewegung (II)

- $V_n = -b_\kappa$  in (3) =>

$$\Phi_t = b_\kappa |\nabla\Phi| \quad (4)$$

- $\Phi$  Distanzfunktion mit Vorzeichen => „Wärmegleichung“

$$\Phi_t = b\Delta\Phi \quad (5)$$

mit  $\Phi$  Temperatur und  $b$  thermale Konduktivität

- (4) und (5) haben denselben Effekt auf die Grenzfläche, solange  $\Phi$  eine Distanzfunktion mit Vorzeichen ist
- Keine Veränderung der Lage der Grenzfläche, sondern Änderung der implizit einbettenden Funktion

# Numerische Diskretisierung

- Bis jetzt: hyperbolische Gleichungen
  - Informationen bewegen sich nur in Richtung der Charakteristika
- parabolische Gleichungen
  - Diskretisierung mit Hilfe des central differencing notwendig, da das Abhängigkeitsgebiet Informationen aller räumlichen Richtungen enthält

## ■ Vorgehen:

- Gleichung (5) sei gegeben
- Central differencing 2.Ordnung

$$\Phi_{xx} \approx (\Phi_{i+1} - 2\Phi_i + \Phi_{i-1}) / (\Delta x)^2$$

$$\Delta\Phi = \Phi_{xx} + \Phi_{yy} + \Phi_{zz}$$

- Zeitschrittrestriktion beim central differencing der 2. Ableitung von  $\Phi$  kombiniert mit der Eulerschen Vorwärts-Zeitdiskretisierung

$$\Delta t((2b / (\Delta x)^2) + (2b / (\Delta y)^2) + (2b / (\Delta z)^2)) < 1 \quad \Delta t = O((\Delta x)^2)$$

- Verringerung der hohen Anforderungen durch Verwendung eines Löser gewöhnlicher Differentialgleichungen mit großen stabilen Regionen

## Numerische Diskretisierung (II)

### ■ Vorgehen (Fortsetzung):

- Anwendung der Eulerschen Rückwärtsgleichung 1.Ordnung auf (5)

$$(\Phi^{n+1} - \Phi^n) / \Delta t = b \Delta \Phi^{n+1}$$

- Fehlerterm  $\Delta t = O(\Delta x)$
- Verbesserung der fehlenden Genauigkeit mit der Trapezregel

$$(\Phi^{n+1} - \Phi^n) / \Delta t = b((\Delta \Phi^n + \Delta \Phi^{n+1}) / 2)$$

$$\Rightarrow O((\Delta x)^2) \text{ und } O((\Delta t)^2)$$

### ■ Kombination von central differencing und Trapezregel:

#### **Crank-Nicolson Schema**

## Konvektion-Diffusions Gleichung

- Folgende Gleichungen sind auswechselbar, wenn  $\Phi$  eine Distanzfunktion mit Vorzeichen ist:

$$\Phi_t + \vec{V} \nabla \Phi = b \Delta \Phi$$

$$\Phi_t + \vec{V} \nabla \Phi = b_\kappa |\nabla \Phi|$$

- Lösung: upwind Methode auf  $\vec{V} \nabla \Phi$  und central differencing auf  $b \Delta \Phi$  bzw.  $b_\kappa |\nabla \Phi|$
- Eine TVD RK Zeitdiskretisierung kann mit einer Zeitschrittrestriktion von 
$$\Delta t((|u|/\Delta x) + (|v|/\Delta y) + (|w|/\Delta z) + (2b/(\Delta x)^2) + (2b/(\Delta y)^2) + (2b/(\Delta z)^2)) < 1$$
 die an jedem Gitterpunkt erfüllt wird, genutzt werden.
- Verbesserung: Ersetzung von  $b$  (mit  $O(1)$ ) durch  $\varepsilon$  (mit  $O(\Delta x)$ ), welches für  $\Delta x \rightarrow 0$  verschwindet

$$\Phi_t + \vec{V} \nabla \Phi = \varepsilon \Delta \Phi$$

# Gliederung

- Bewegungen in extern erzeugten Geschwindigkeitsfeldern
  - Hilfreiche Methoden
    - Konvektion
    - Upwind Differencing
    - Hamilton-Jacobi Essentially Nonoscillatory (ENO) Polynomial Interpolation
    - Hamilton-Jacobi Gewichtete ENO (WENO)
    - TVD (Total Variation Diminishing) Runge Kutta Methode
  
- Bewegungen in intern erzeugten Geschwindigkeitsfeldern
  - Gleichungen
    - Bewegungsgleichung
    - Wärmeleichung
  - Numerische Diskretisierung
  
- Hamilton-Jacobi Gleichungen
  - Erhaltungsgesetz
  - Eulersche Vorwärtszeitdiskretisierung

# Hamilton-Jacobi Gleichungen

$$\Phi_t + H(\nabla\Phi) = 0$$

(6)

## Beispiel:

Konvektion in einem extern erzeugten Geschwindigkeitsfeld

$$H(\nabla\Phi) = \vec{V} \nabla\Phi$$

Level Set Gleichung

$$H(\nabla\Phi) = V_n |\nabla\Phi|$$

# Erhaltungsgesetz

- Eindimensionales Erhaltungsgesetz

$$u_t + F(u)_x = 0$$

$u$  erhaltene Masse  
 $F(u)$  Flussfunktion

- Beispiele

- Kontinuitätsgleichung

$$\delta_t + (\delta w)_x = 0$$

$\delta$  Dichte des Materials  
 $w$  Geschwindigkeitsfeld des Fluides

- Burger Gleichung

$$u_t + (u^2 / 2)_x = 0$$

skalares Erhaltungsgesetz

## Erhaltungsgesetz (II)

- Erweiterung der numerischen Methoden der Burger Gleichung zur Lösung ein- und mehrdimensionaler Euler-Gleichungen der Gasdynamik

Hamilton-Jacobi Gleichung:

$$\Phi_t + H(\Phi_x) = 0$$

$$\Rightarrow (\Phi_x)_t + H(\Phi_x)_x = 0$$

Setze  $u = \Phi_x$

$$\Rightarrow u_t + H(u)_x = 0$$

- Direkte Korrespondenz im eindimensionalen zwischen Hamilton-Jacobi Gleichung und Erhaltungsgesetz
  - => Ableitung vieler nützlicher Eigenschaften
- Diese Analogie schlägt im mehrdimensionalen fehl!
  - Verwirklichung: dimension-by-dimension Technik

# Eulersche Vorwärtszeitdiskretisierung

- Forward-Euler Zeitdiskretisierung einer Hamilton-Jacobi Gleichung:

$$\left( (\Phi^{n+1} - \Phi^n) / \Delta t \right) + \hat{H}^n(\Phi_x^-, \Phi_x^+, \Phi_y^-, \Phi_y^+, \Phi_z^-, \Phi_z^+) = 0 \quad (7)$$

$\hat{H}$  ist:

- numerische Approximation von  $H(\Phi_x, \Phi_y, \Phi_z)$
- numerische Hamiltonian
- konsistent  $\hat{H}(\Phi_x, \Phi_x; \Phi_y, \Phi_y; \Phi_z, \Phi_z) = H(\Phi_x, \Phi_y, \Phi_z)$

## Werkzeug

- Diskretisierung räumlicher Ableitungen mit one-sided differencing 1.Ordnung oder mit HJ ENO/HJ WENO Schemas höherer Ordnung
- Monotone Schematas
- Direkte Erweiterung von (7) zu TVD Runge-Kutta höherer Ordnung
- CFL-Bedingung

$$\Delta t \max \{ (H_1 / \Delta x) + (H_2 / \Delta y) + (H_3 / \Delta z) \} < 1$$

mit  $H_1, H_2$  und  $H_3$  partielle Ableitungen von  $H$  in Bezug auf  $\Phi_x, \Phi_y, \Phi_z$ .

## Numerische Diskretisierung (II)

- Numerische Diskretisierung mit Hilfe eines Schemas
- Im Folgenden Vorstellung von 3 Schemata:
  - Lax-Friedrich Schema
  - Roe-Fix Schema
  - Schema von Godunov

## Lax-Friedrich Schema

$$\hat{H} = H\left(\frac{\Phi_x^- + \Phi_x^+}{2}, \frac{\Phi_y^- + \Phi_y^+}{2}\right) - \alpha^x \left(\frac{\Phi_x^- - \Phi_x^+}{2}\right) - \alpha^y \left(\frac{\Phi_y^- - \Phi_y^+}{2}\right) \quad (8)$$

- $\alpha^x, \alpha^y$  Zerteilungskoeffizienten, die die Menge der numerischen Zähflüssigkeiten kontrollieren

$$\alpha^x = \max |H_1(\Phi_x, \Phi_y)|$$

$$\alpha^y = \max |H_2(\Phi_x, \Phi_y)|$$

- Auswahl des Maximums über die Ganze zu berechnende Region
  - Identifizierung der max und min Werte von  $\Phi_x$ , mit Betrachtung aller Werte von  $\Phi_x^+$  und  $\Phi_x^-$  des kartesischen Netzes  $\Rightarrow I^x = [\Phi_x^{\min}, \Phi_x^{\max}]$
  - Analog für  $I^y$
  - Setzen der Koeffizienten  $\alpha^x, \alpha^y$  auf die Maximumswerte von  $|H_1(\Phi_x, \Phi_y)|$  und  $|H_2(\Phi_x, \Phi_y)|$  mit  $\Phi_x \in I^x$  und  $\Phi_y \in I^y$

## Lax-Friedrich Schema (II)

- Eindämmung der künstlichen Streuung durch Wahl des Minimums von  $\max(\alpha)$
- Wahl von  $\alpha^x$  als größten Wert von  $|u|$  und  $\alpha^y$  von  $|v|$  mit Hilfe der LF Methode (Nutzung des Wertes in Regionen mit hoher, wie mit niedriger Geschwindigkeit)
  - Gute Lösung bei hoher Geschwindigkeit
  - Schlechte Lösung mit vielen numerischen Fehlern bei geringer Geschwindigkeit
  - Lösung „*Stencil-Lax Friedrichs Schema (SLF)*“ :  
Bestimmung von  $\alpha$  nur durch Netzpunkte, die nahe bei  $\vec{x}_{i,j}$  sind
    - Faustregel  
 $\vec{x}_{i-3,j}$  bis  $\vec{x}_{i+3,j}$  in x-Richtung und  
 $\vec{x}_{i,j-3}$  bis  $\vec{x}_{i,j+3}$  in y-Richtung wird in lokale Nachbarschaftssuche einbezogen um  $\alpha$  zu bestimmen.

## Lax-Friedrich Schema (III)

### □ *Local Lax-Friedrich (LLF) Schema:*

- Bestimmung von  $I^x$  an jedem Netzpunkt, unter Verwendung der Werte von  $\Phi_x^+$  und  $\Phi_x^-$  an diesem speziellen Netzpunkt
- Bestimmung von  $\alpha^x$
- Bestimmung von  $I^y$  weiterhin nach LF oder SLF Schema
- Analoges Vorgehen für  $\alpha^y$

### □ *Local Local Lax-Friedrich (LLLF) Schema:*

- Geringe numerische Streuung
- Bestimmung von  $I^x$  an jedem Netzpunkt unter Verwendung der Werte von  $\Phi_x^+$  und  $\Phi_x^-$  an diesem Netzpunkt
- Bestimmung von  $I^y$  analog  
=> Bestimmung von  $\alpha^x$  und  $\alpha^y$

### ■ **Bemerkungen**

- Falls H trennbar: Reduzierung von LLLF zu LLF
- LLF monotones Schema

## Roe-Fix Schema

$$\hat{H} = H(\Phi_x^*, \Phi_y^*) - \alpha^x ((\Phi_x^+ - \Phi_x^-) / 2) - \alpha^y ((\Phi_y^+ - \Phi_y^-) / 2)$$

$$\text{mit: } \alpha^x = \alpha^y = 0$$

(9)

- Durch Nutzung der Knotenwerte von  $\Phi_x^\pm$  und  $\Phi_y^\pm$  analog zum LLLF Schema => Bestimmung von  $I^x$  und  $I^y$
- Upwinding anwendbar, wenn  $H^1(\Phi_x, \Phi_y)$  dasselbe Vorzeichen  $\forall \Phi_x \in I_x$  und  $\forall \Phi_y \in I_y$  hat  
=> Bestimmung der Fließrichtung der Informationen möglich
- **Upwinding**
  - $H^1$  und  $H^2$  kein Vorzeichenwechsel
    - $\alpha^x = \alpha^y = 0$
    - $H^1 > 0$  Informationen gehen von links nach rechts:  $\Phi_x^* = \Phi_x^-$
    - $H^1 < 0$   $\Phi_x^* = \Phi_x^+$
    - $H^2$  analog
  - $H^1$  und  $H^2$  Vorzeichenwechsel
    - Eigenwert= 0
    - Wechsel zum LLF Schema, wegen benötigter künstlicher Zähflüssigkeit
    - Dämpfung in die Richtung, wo es Schallpunkte gibt genügt => Definition von  $I^x$  und  $I^y$  mit Hilfe der LLF Methode

## Roe-Fix Schema (II)

### ■ Weiteres Vorgehen:

- Falls  $H_1(\Phi_x, \Phi_y)$  die Vorzeichen  $\forall \Phi_x \in I_{LLF}^x$  und  $\forall \Phi_y \in I_{LLF}^y$  nicht ändert:

$$\Phi_x^* = \Phi_x^- \text{ bzw. } \Phi_x^+ \quad (\text{abhängig vom Vorzeichen von } H_1)$$

$$\alpha^x = 0$$

=> Schallpunkte in  $H_2$

$$\Phi_y^* = ((\Phi_y^- + \Phi_y^+) / 2) \quad \alpha^y \text{ nach LLF Schema}$$

- Falls  $H_2(\Phi_x, \Phi_y)$  die Vorzeichen  $\forall \Phi_x \in I_{LLF}^x$  und  $\forall \Phi_y \in I_{LLF}^y$  nicht ändert:  
Vorgehen analog zu  $H_1(\Phi_x, \Phi_y)$

- Falls  $H_1$  und  $H_2$  sich ändernde Vorzeichen haben, so existieren in beiden Richtungen Schallpunkte; Fortsetzung mit dem Standard LLF Schema am Gitterpunkt

- Man kann nicht bestimmen, ob upwinding genutzt werden kann, ohne  $\Phi_x^\pm$  und  $\Phi_y^\pm$  zu bestimmen

# Schema von Godunov

$$\hat{H} = \text{ext}_x \text{ext}_y H(\Phi_x, \Phi_y)$$

(10)

- Falls  $\Phi_x^- < \Phi_x^+ : \text{ext}_x H = \min(H) \forall \Phi_x \in I^x$   
 $\Phi_x^- > \Phi_x^+ : \text{ext}_x H = \max(H) \forall \Phi_x \in I^x$   
 $\Phi_x^- = \Phi_x^+ : \text{ext}_x H = \Phi_x^- (= \Phi_x^+)$
- Analog für  $\Phi_y$
- Achtung:  $\text{ext}_x \text{ext}_y H \neq \text{ext}_y \text{ext}_x H$
- Identisch, falls H trennbar: So gilt  $\text{ext}_x \text{ext}_y H = \text{ext}_x (u\Phi_x) + \text{ext}_y (v\Phi_y)$

$$\Phi_x^- < \Phi_x^+ \Rightarrow \text{Bestimme } \min(u\Phi_x)$$

$$u > 0 : \Phi_x^-$$

$$u < 0 : \Phi_x^+$$

$$u = 0 : u\Phi_x = 0$$

$$\Phi_x^- > \Phi_x^+ \Rightarrow \text{Bestimme } \max(u\Phi_x)$$

$$u > 0 : \Phi_x^-$$

$$u < 0 : \Phi_x^+$$

$$u = 0 : u\Phi_x = 0$$

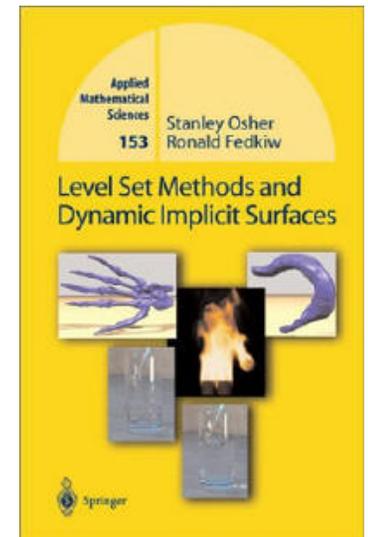
$$\Phi_x^- = \Phi_x^+ \Rightarrow u\Phi_x \text{ eindeutig bestimmt}$$

# Zusammenfassung

- Hilfsmittel zur Lösung der Level Set Gleichung bei extern erzeugten Geschwindigkeitsfeldern
  - Upwind Differencing
  - HJ ENO
  - HJ WENO
  - TVD RK Methode
  
- Hilfsmittel zur Lösung der Level Set Gleichung bei intern erzeugten Geschwindigkeitsfeldern
  - Numerische Diskretisierung
    - Central Differencing
    - Trapezregel
  
- Verallgemeinerung der Ansätze: Hamilton-Jacobi Gleichung
  - Eulersche Vorwärtszeitdiskretisierung
    - Lax-Friedrich Schema
    - Roe-Fix Schema
    - Schema von Godunov

## Literaturverzeichnis

- S. Osher, R. Fedkiw *"Level set methods and dynamic implicit surfaces"*, Springer, 2003.
- W.Press, S.Teukolsky, W.Vetterling, B.Flannery *"Numerical Recipes in C++"*, Cambridge University Press, 1988-1992
- [http://egartner.de/num\\_claw.pdf](http://egartner.de/num_claw.pdf)
- <http://www.wikipedia.de>
- [http://www.bic.mni.mcgill.ca/users/maudette/fem\\_home.html](http://www.bic.mni.mcgill.ca/users/maudette/fem_home.html)
- [http://www.ualberta.ca/dept/math/gauss/fcm/calculus/multvrbl/basic/ImplctFunctns/implct\\_functn\\_illstrtn1.htm](http://www.ualberta.ca/dept/math/gauss/fcm/calculus/multvrbl/basic/ImplctFunctns/implct_functn_illstrtn1.htm)





**Vielen Dank für Ihre  
Aufmerksamkeit!**

**Gibt es noch Fragen?**