# Eulersche Fluid Simulation für die Computergrafik

## Scientific Computing

Tim Krüger

Hochschule Niederrhein
Fachbereich Elektrotechnik & Informatik

06. November 2024

Hochschule Niederrhein
University of Applied Sciences
Elektrotechnik und Informatik
Faculty of Electrical Engineering and Computer Science

# Motivation

- Realistische, echtzeitfähige Flüssigkeitsbewegungen für Computergrafik-Anwendungen
  - Visualisierungen, Computerspiele etc.
- Zeitlimitierung von 16.6 ms (**für alle Prozesse**) wenn Framerate von 60 fps angestrebt wird
  - Inputmanagement, UI, Rendering, Physics ...
- Komplexes, nicht-triviales Problem - *"Rabbithole"*
  - Physik ist kompliziert - tiefe Theorie
  - Implementierung (performant) - praktische Herausforderung
  - Numerische Einschränkungen
- Deswegen: Vereinfachungen und Annahmen (nächste Folie)
  - Keine Forschungsqualität
  - Meistens nicht mal double-precision



Hochschule Niederrhein
University of Applied Sciences
Elektrotechnik und Informatik
Faculty of Electrical Engineering and Computer Science

# Allgemeines

- Als *Fluid* bezeichnet man sowohl eine Flüssigkeit als auch ein Gas
  - Ähnliche physikalische Struktur, dieselbe Simulationsmethode
- 1. **Annahme:** Inkompressibles Fluid
  - Gute Approximation für Wasser
  - 10.000 $kg/cm^2$ ($\approx$ 10.000 bar) führen zu nur 3% Kompression
  - Auch für freie Gase geeignet
- 2. **Annahme:** Viskosität von 0
  - Zähigkeit der Flüssigkeit (Wasser vs: Honig)
  - Realistische Vereinfachung für Gase und Wasser
- Simulation würde ansonsten deutlich komplizierter werden
  - Lösung der vollständigen Navier-Stokes-Gleichungen



Hochschule Niederrhein
University of Applied Sciences
Elektrotechnik und Informatik
Faculty of Electrical Engineering and Computer Science

# Navier-Stokes Gleichungen (1)

- Mathematisches Modell der Strömung von linear-viskosen newtonschen Fluiden
  - Über Jahrzehnte entwickelt (ungefähr 1822-1850)
  - Benannt nach Claude-Louis Navier und George Gabriel Stokes
- Satz von partiellen Differentialgleichungen
- Teil der sieben Millenium-Probleme: Nicht bekannt, ob es für den allgemeinen 3D Fall eine überall definierte, glatte (eindeutige) Lösung gibt
  - Für 2D wissen wir es
- Im folgenden wird immer von der inkompressiblen Form der Navier-Stokes Gleichungen ausgegangen
  - Volumen bzw. Dichte der Flüssigkeit bleibt immer gleich

# Differentialgleichungen

Differentialgleichungen beschreiben die Beziehung zwischen einer Funktion und ihren Ableitungen in Bezug auf eine oder mehrere unabhängige Variablen.

- Viele Naturgesetze können mittels Differentialgleichungen formuliert werden
- Den Grad der höchsten Ableitung nennt man Ordnung

Anhand der Anzahl der unabhängigen Variablen unterscheidet man:

- Gewöhnliche Differentialgleichungen (engl. ODE)
  - Eine unabhängige Variable
- Partielle Differentialgleichungen (engl. PDE)
  - Mehrere unabhängige Variablen

# Gewöhnliche Differentialgleichungen

Bei gewöhnlichen Differentialgleichungen (ODE) hängt die gesuchte Funktion $y(x)$ von nur einer Variablen ab.

- Nur gewöhnliche Ableitungen der gesuchten Funktion nach der unabhängigen Variable treten auf
- Beispiel: $\frac{dy}{dx} = ky$ (Exponentielles Wachstum)

Man unterscheidet gewöhnliche Differentialgleichungen des Weiteren in *implizit*, *explizit*, *linear* und *nichtlinear*.

- Die Methoden zur Lösung variieren stark und hängen von der Art der Differentialgleichung ab
- Kann die ODE explizit gelöst werden, lässt sich eine analytische Formel für die Lösung $y(x)$ finden
- Viele ODEs, besonders nichtlineare, erfordern jedoch numerische Methoden zur Approximation einer Lösung



Hochschule Niederrhein
University of Applied Sciences
Elektrotechnik und Informatik
Faculty of Electrical Engineering and Computer Science

# Partielle Differentialgleichungen

Bei partiellen Differentialgleichungen (PDE) hängt die gesuchte Funktion von mehreren Variablen ab und es treten partielle Ableitungen nach mehr als einer Variable auf.

- Die Theorie ist mathematisch nicht abgeschlossen, sondern Gegenstand der aktuellen Forschung
  - Die Lösungstheorie linearer partieller Differentialgleichungen ist größenteils erforscht
  - Die Lösungstheorie nichtlinearer partieller Differentialgleichungen dagegen enthält noch viele Lücken
- Gleichungen höherer Ordnung sind i.d.R. schwieriger zu lösen
- Meist nur numerische Lösung möglich (Finite-Elemente, ...)
- Anzahl der Variablen $\rightarrow$ Untersuchung theor. Lösbarkeit:
  - Navier-Stokes-Gleichungen in 2D: Beweis von Existenz-, Eindeutigkeits- und Regularitätsaussagen
  - Navier-Stokes-Gleichungen in 3D: **Offen** - Teil der Millennium-Probleme



Hochschule Niederrhein
University of Applied Sciences
Elektrotechnik und Informatik
Faculty of Electrical Engineering and Computer Science

# Navier-Stokes Gleichungen (2)

1.1 **Inkompressibilität** $\nabla \cdot \vec{u} = 0$

1.2 **Momentum** $\rho \left( \frac{\partial \vec{u}}{\partial t} + \vec{u} \cdot \nabla \vec{u} \right) = -\nabla p + v \nabla^2 \vec{u} + \vec{g}$



Hochschule Niederrhein
University of Applied Sciences
Elektrotechnik und Informatik
Faculty of Electrical Engineering and Computer Science

# Inkompressibilitätsgleichung (1)

**Inkompressibilität** $\nabla \cdot \vec{u} = 0$

Wird auch als Kontinuitätsgleichung oder *Conservation of Mass*-Gleichung bezeichnet.

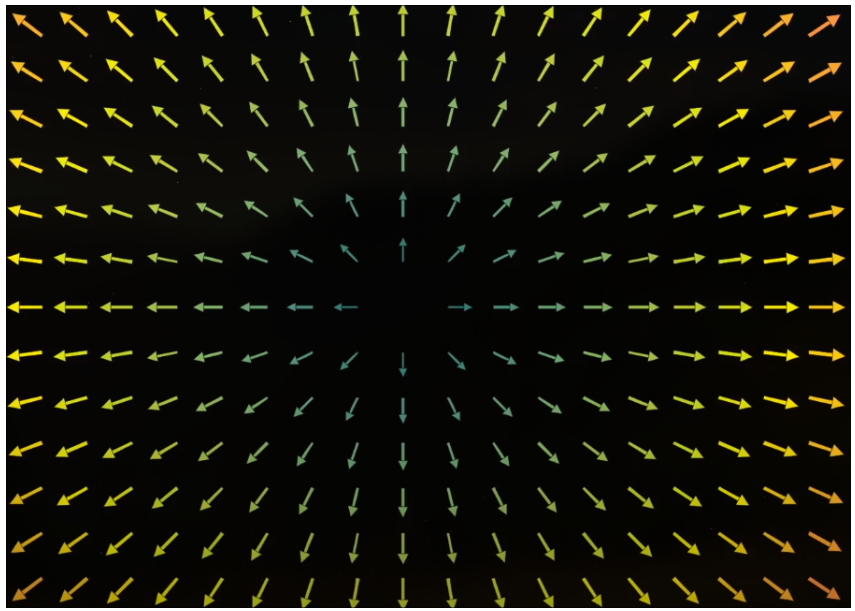
- $\vec{u}$ := Geschwindigkeitsfeld (engl. velocity field)
  - Velocity := Speed with direction $(u, v, w)$
- $\nabla$ := Divergenzoperator
  - Summe der einzelnen Komponenten der partiellen Ableitungen des Vektorfeldes entlang der jeweiligen Achsen
  - Ergebnis ist ein skalares Feld welches die Divergenz des Vektorfeldes an jedem Punkt beschreibt
  - Wie stark kontrahieren oder divergieren die Vektoren an diesem Punkt?

# Inkompressibilitätsgleichung (2)

$$\nabla \cdot \vec{u} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0$$

Die Gleichung sagt also aus, dass die Divergenz des Geschwindigkeitfeldes Null sein muss.

- Man bezeichnet ein solches Vektorfeld als *inkompressibel* oder *divergenzfrei*



# Impulsgleichung (1)

Wird auch als *Erhaltung des Momentums* bezeichnet.

- Fangen wir mit der gegebenen Definition aus 1.2 in etwas vereinfachter Form an:

$$\rho \frac{D\vec{u}}{Dt} = -\nabla p + v\nabla^2 \vec{u} + \vec{g}$$

- Hinter dieser Gleichung verbirgt sich nichts anderes als *Newtons* 2. Gesetz: $\vec{F} = m\vec{a}$

$$\underbrace{\rho}_{\text{Masse}} \quad \underbrace{\frac{D\vec{u}}{Dt}}_{\text{Beschleunigung}} = \underbrace{-\nabla p + v\nabla^2 \vec{u}}_{\text{Interne Kräfte}} + \underbrace{\vec{g}}_{\text{Externe Kräfte}}$$



Hochschule Niederrhein
University of Applied Sciences
Elektrotechnik und Informatik
Faculty of Electrical Engineering and Computer Science

# Impulsgleichung (2)

- $\rho :=$ **Masse** bzw. Dichte (engl. density)
  - Wasser $:= 1000\ kg/m^3$
  - Luft $:= 1.3\ kg/m^3$
- Die **Beschleunigung** geschrieben als materielle Ableitung:

$$\frac{D\vec{u}}{Dt} = \frac{\partial \vec{u}}{\partial t} + \vec{u} \cdot \nabla \vec{u}$$

- $\frac{\partial \vec{u}}{\partial t} :=$ Lokale Änderung an einem Ort ohne Fluidbewegung
- $\vec{u} \cdot \nabla \vec{u} :=$ Konvektive Änderung durch Bewegung des Fluids
  - Entspricht dem Produkt des Geschwindigkeitsfeldes $\vec{u}$ und dessen Gradienten
  - Die Berechnung erfolgt komponentenweise und das Ergebnis ist ein Vektorfeld
- Insgesamt: Änderung eines "Partikels", welches sich mit der Strömung bewegt



Hochschule Niederrhein
University of Applied Sciences
Elektrotechnik und Informatik
Faculty of Electrical Engineering and Computer Science

# Impulsgleichung (3)

- **Interne Kräfte** (Kräfte welche innerhalb des Fluids wirken):

$$-\nabla p + v\nabla^2 \vec{u}$$

- $\nabla :=$ Gradient
  - Richtung und Stärke der "steilsten" Steigung:

$$\nabla = \left( \frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} \right)$$

- $p :=$ Druck (engl. pressure)
  - Kraft pro Flächeneinheit, welche sich auf das Fluid auswirkt
  - $-\nabla p :=$ Bewegung des Fluids in Richtung niedrigeren Drucks
- $v :=$ Viskosität (engl. viscosity)
  - Wie stark verformt sich das Fluid während es fließt?
  - $v\nabla^2 \vec{u} :=$ Modelliert viskose Effekte; Ausgleich von Geschwindigkeitsunterschieden



Hochschule Niederrhein
University of Applied Sciences
Elektrotechnik und Informatik
Faculty of Electrical Engineering and Computer Science

# Impulsgleichung (4)

- $\nabla^2$ := Laplace-Operator (engl. Laplacian)
  - Verschiedene Schreibweisen: $\nabla^2, \nabla \cdot \nabla$ oder auch $\Delta$
  - Differentialoperator zweiter Ordnung, entspricht der Divergenz des Gradienten:

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} + \frac{\partial^2 f}{\partial z^2}$$

- $\nu \nabla^2 \vec{u}$:

1. Bilde den Laplacian des Vektorfeldes $\vec{u}$. Der Laplace-Operator wird dabei auf jede Komponente des Vektorfeldes $\vec{u}$ einzeln angewendet, wodurch wieder ein Vektorfeld entsteht
2. Multipliziere das resultierende Vektorfeld mit der Viskosität $\nu$



Hochschule Niederrhein
University of Applied Sciences
Elektrotechnik und Informatik
Faculty of Electrical Engineering and Computer Science

# Impulsgleichung (5)

- **Externe Kräfte** (Alle Kräfte, welche von außerhalb auf die Flüssigkeit einwirken):
  - $\vec{g}$ := Gravitation (engl. gravity)
    - $9.81\,\mathrm{m/s^2}$
  - Zusätzliche Kräfte, die zur Erzeugung von Fluid-Bewegungen beitragen, werden ebenfalls in $\vec{g}$ zusammengefasst

Bei vielen Problemen ist das Fluid nicht nur inkompressibel, sondern hat auch konstante Dichte. Dann vereinfacht sich Gleichung 1.2 zu:

$$\frac{D\vec{u}}{Dt} = -\nabla p + \nu \nabla^2 \vec{u} + \vec{g}$$

# Euler-Gleichungen

Die Navier-Stokes-Gleichungen ohne viskose Effekte werden *Euler-Gleichungen* genannt:

$$\nabla \cdot \vec{u} = 0 \tag{1.3}$$

$$\frac{D\vec{u}}{Dt} + \frac{1}{\rho}\nabla p = \vec{g} \tag{1.4}$$

- Beschreiben "ideales" Fluid, welches keine inneren Reibungseffekte aufweist
- Eignen sich gut für diskrete Approximationen und eine numerische Umsetzung
- Bei konstanter Dichte vereinfacht sich 1.4 entsprechend weiter

# Numerische Lösung (1)

Es existieren zwei prominente Ansätze zur Fluid-Simulation:

- *Lagrange*: Bewegung einzelner Partikel im Raum
- *Euler*: Strömung an festen Punkten eines Gitters

Zur numerischen Berechnung des Gradienten wird jede Komponente durch einen Differenzenquotienten ersetzt.

$$\text{Vorwärtsdifferenz: } f'(x) \approx \frac{f(x+h) - f(x)}{h} + O(h)$$

$$\text{Rückwärtsdifferenz: } f'(x) \approx \frac{f(x) - f(x-h)}{h} + O(h)$$

$$\text{Zentrale Differenz: } f'(x) \approx \frac{f(x+h) - f(x-h)}{2h} + O(h^2)$$

- $h :=$ Kleiner positiver Wert (Schrittweite)
- $O(h) :=$ Wachstum des Fehlers



Hochschule Niederrhein
University of Applied Sciences
Elektrotechnik und Informatik
Faculty of Electrical Engineering and Computer Science

# Numerische Lösung (2)

- Intuition: Approximation der Ableitung indem benachbarte Gitterpunkte verglichen werden
- *Zentrale Differenz* bietet generell ein höhere Genauigkeit und wird deshalb oft bevorzugt
- Das Hauptproblem bei der Nutzung von Differenzenquotienten ist die Wahl der Schrittweite $h$
  - Ist $h$ zu klein, kann es zu numerischen Stabilitätsproblemen und signifikanten Rundungsfehlern kommen (Auslöschung)
  - $h$ darf deswegen von der Maschinengenauigkeit abhängige Schranken nicht unterschreiten
  - Ist $h$ zu groß, wächst der Fehler linear bzw. quadratisch



Hochschule Niederrhein
University of Applied Sciences
Elektrotechnik und Informatik
Faculty of Electrical Engineering and Computer Science

# Staggered Grid (1)

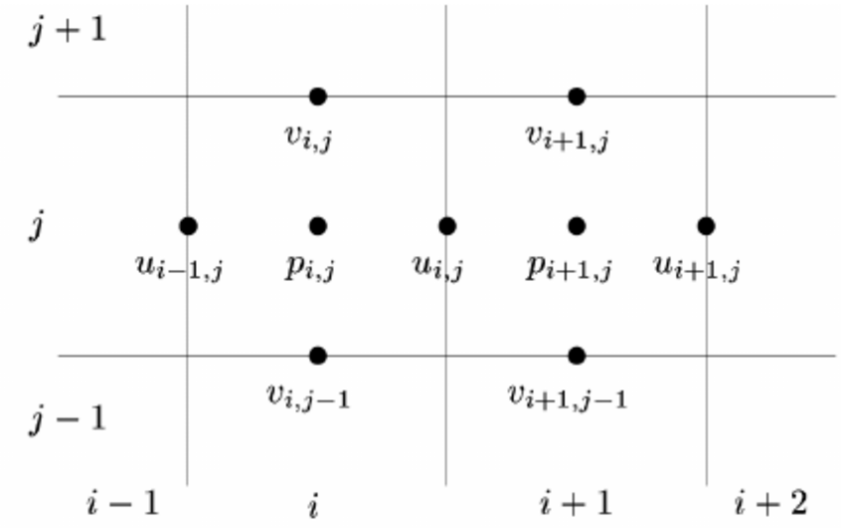
Optimierte Gitterstruktur für das Geschwindigkeitsfeld:

- Wird auch als *MAC-Grid* bezeichnet
- Geschwindigkeit ist ein 2D-Vektor: $\vec{v} = \begin{pmatrix} x \\ y \end{pmatrix}$
- Der Wert einer jeden Gitterzelle $(i, j)$ wird im Zentrum der Zelle gesampled
- Die horizontale $u$-Komponente wird im Zentrum der vertikalen Zell-Faces gesampled
  - $u_{i+1/2,j}$ für die $u$-Geschwindigkeit zwischen $(i, j)$ und $(i + 1, j)$
- Die vertikale $v$-Komponente wird im Zentrum der horizontalen Zell-Faces gesampled
  - $v_{i,j+1/2}$ für die $v$-Geschwindigkeit zwischen $(i, j)$ und $(i, j + 1)$



Hochschule Niederrhein
University of Applied Sciences
Elektrotechnik und Informatik
Faculty of Electrical Engineering and Computer Science

# Staggered Grid (2)




- Staggered Grids erlauben für eine bessere (numerische) Ableitungsformel (ohne *null-space-problem*):

$$\left(\frac{\partial q}{\partial x}\right)_i \approx \frac{q_{i+1/2} - q_{i-1/2}}{\Delta x}$$



Hochschule Niederrhein
University of Applied Sciences
Elektrotechnik und Informatik
Faculty of Electrical Engineering and Computer Science

# Simulationsaufbau

1. Kräfte hinzufügen (modifiziere Geschwindigkeitsfeld)
2. Mache das Fluid inkompressibel (*Projection*)
3. (Optional) Extrapoliere für Randpunkte
4. Bewege das Geschwindigkeitsfeld (*Advection*)
5. (Optional) Visualisiere den Fluss



Hochschule Niederrhein
University of Applied Sciences
Elektrotechnik und Informatik
Faculty of Electrical Engineering and Computer Science

# Kräfte hinzufügen

for all i, j

$$v_{i,j} \leftarrow v_{i,j} + \Delta t \cdot g$$

- $g :=$ Gravity
- $\Delta t :=$ Zeitschritt

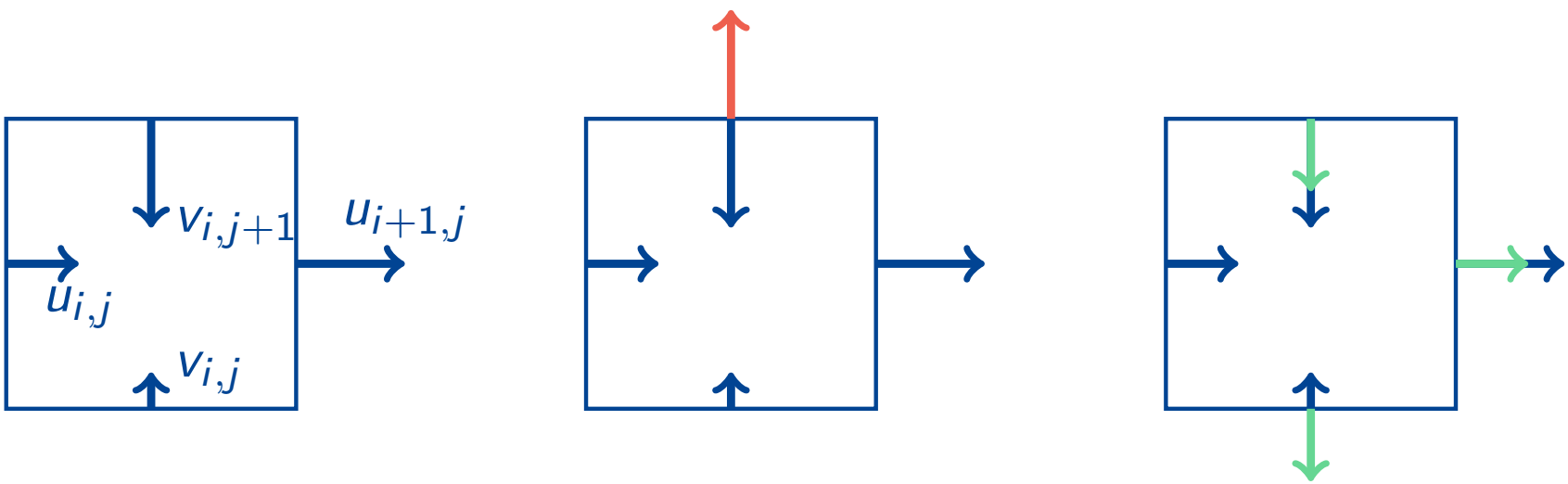


Hochschule Niederrhein
University of Applied Sciences
Elektrotechnik und Informatik
Faculty of Electrical Engineering and Computer Science

# Projection (1)

Modifiziert die Geschwindigkeiten um $\vec{u}$ divergenzfrei zu machen und hält Randwertkonditionen ein.

- Alle Zellseiten müssen mit demselben Wert verrechnet werden





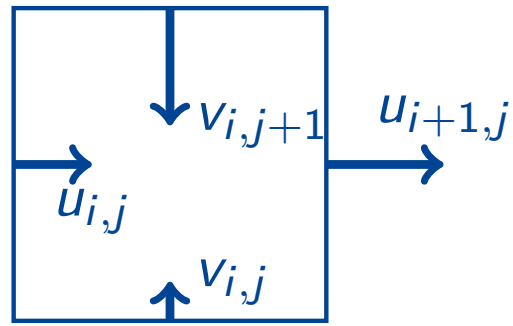

Hochschule Niederrhein
University of Applied Sciences
Elektrotechnik und Informatik
Faculty of Electrical Engineering and Computer Science

# Projection (2)

Berechnung des Druckwertes mit dem die Geschwindigkeiten der einzelnen Zellseiten verrechnet werden.

$$d \leftarrow u_{i+1,j} - u_{i,j} + v_{i,j+1} - v_{i,j}$$

$$\begin{aligned}
u_{i,j} &\leftarrow u_{i,j} + d/4 \\
u_{i+1,j} &\leftarrow u_{i+1,j} - d/4 \\
v_{i,j} &\leftarrow v_{i,j} + d/4 \\
v_{i,j+1} &\leftarrow v_{i,j+1} - d/4
\end{aligned}$$



# Projection (3)

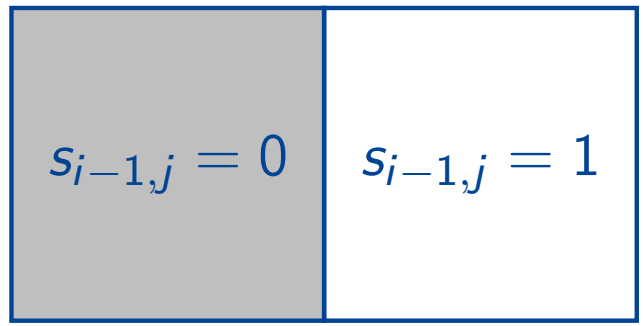
Berechnung des Gitters:

- Randzellen werden einfach auf 0 gesetzt
- Anzahl an benachbarten Randzellen muss beachtet werden
- Lösung mittels Gauss-Seidel

$$d \leftarrow u_{i+1,j} - u_{i,j} + v_{i,j+1} - v_{i,j}$$

$$s \leftarrow s_{i+1,j} + s_{i-1,j} + s_{i,j+1} - s_{i,j-1}$$

$$\begin{aligned}
u_{i,j} &\leftarrow u_{i,j} + d\, s_{i-1,j}/s \\
u_{i+1,j} &\leftarrow u_{i+1,j} - d\, s_{i+1,j}/s \\
v_{i,j} &\leftarrow v_{i,j} + d\, s_{i,j-1}/s \\
v_{i,j+1} &\leftarrow u_{i,j+1} - d\, s_{i,j+1}/s
\end{aligned}$$






Hochschule Niederrhein
University of Applied Sciences
Elektrotechnik und Informatik
Faculty of Electrical Engineering and Computer Science

# Advection (1)

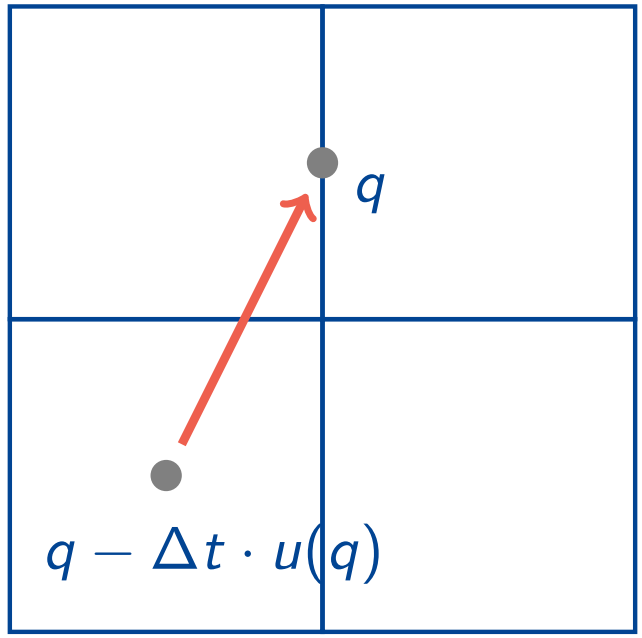
Bewegt das Fluid ($q :=$ generische Quantität) entlang des Geschwindigkeitsfeldes $\vec{u}$ für ein Zeitintervall $\Delta t$.

- Sollte grundsätzlich **nur** auf ein divergenzfreies Geschwindigkeitsfeld angewandt werden
- Es existieren verschiedene Ansätze und Implementierungen

Eine populäre Wahl ist *Semi-Lagrangian* Advection:

- Idee: Man stelle sich ein "Partikel" und dessen Fluss vor
  - Physikalisch-motivierter Ansatz
  - Keine wirklichen Partikel, immernoch statisches Gitter!
- Finde vorherige Position der $u$-Komponente und $v$-Komponente des Geschwindigkeitsfeldes

# Advection (2)




- Diese Updateregel entspricht *Forward-Euler*!



Hochschule Niederrhein
University of Applied Sciences
Elektrotechnik und Informatik
Faculty of Electrical Engineering and Computer Science

# Advection (3)

Updateregel für Forward-Euler auf einem Gitter:

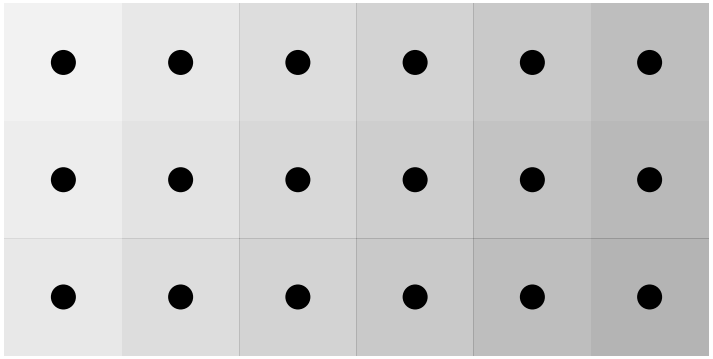
$$q_i^{n+1} = q_i^n - \Delta t \cdot u_i^n \frac{q_{i+1}^n - q_{i-1}^n}{2\Delta x}$$

- Nutzt die *Zentrale Differenz*, um die räumliche Ableitung zu approxieren
  - Vorteil des *Staggered Grid*
- Leider ist Forward-Euler nicht stabil!
  - Lösung: Mehrere Schritte von Forward-Euler mit Mittelung (entspricht *Runge-Kutta2* Verfahren)
- Implementierung folgt ...

# Visualisierung

Visualisierung des Flusses benötigt nichts besonderes, alles notwendige ist bereits im Geschwindigkeitsfeld vorhanden.

- Idee: Speichere einen Dichtewert zwischen 0 und 1 im Zentrum einer jeden Zelle
  - Wird manchmal auch als *Dye*-Wert (Färbung) bezeichnet
- Führe die Advection für diesen Wert durch
  - Dafür wird die Geschwindigkeit im Zentrum einer jeden Zelle genommen: $(u_{i,j} + u_{i+1,j}) * 0.5$ und $(v_{i,j} + v_{i,j+1}) * 0.5$
  - Dann wird (in einer geraden Linie) der Dichtewert an der vorherigen Position interpoliert



# Quellen

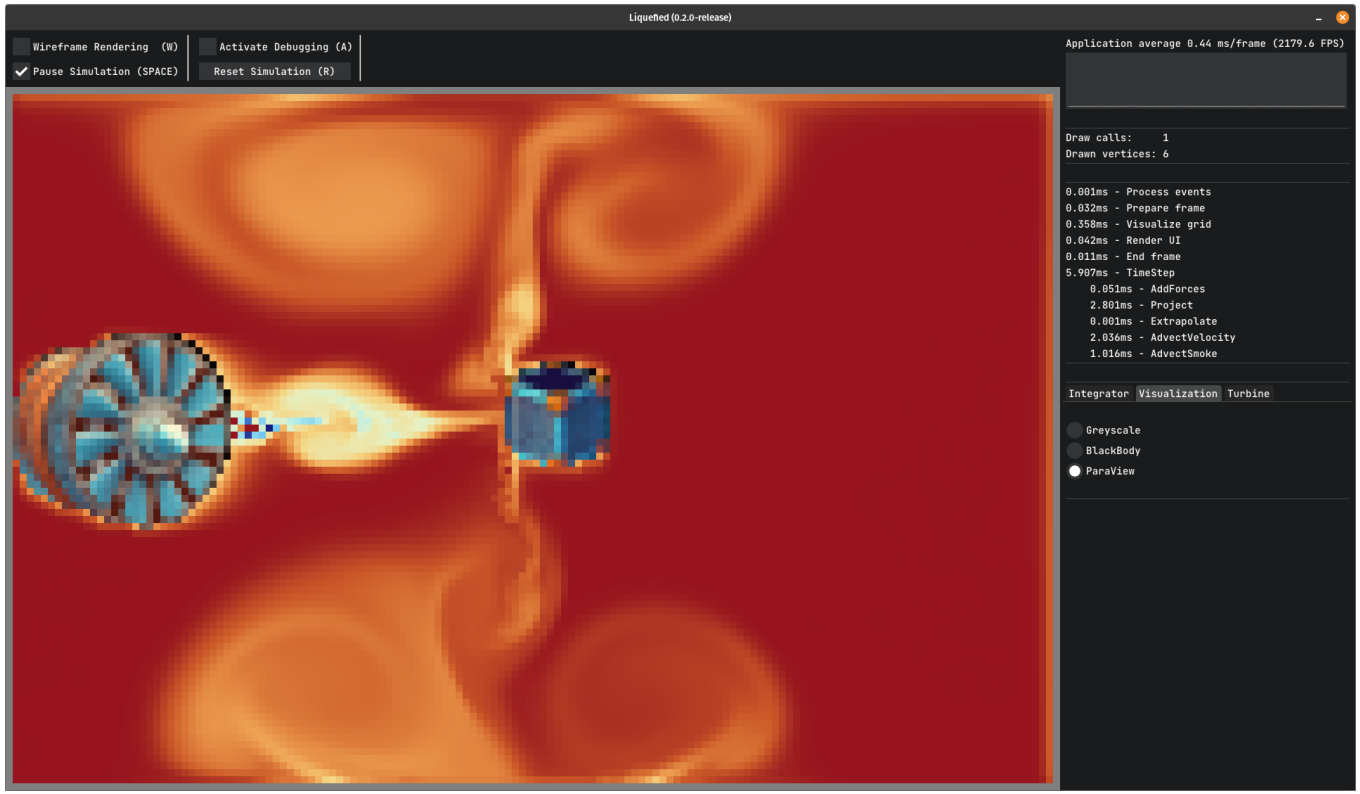

- **Eulerian Fluid Simulator**
  M. Müller, *Ten Minute Physics*, 2022
  matthias-research.github.io/pages/tenMinutePhysics
- **Fluid Simulation for Computer Graphics - 2nd Edition**
  R. Bridson, CRC Press, 2016
- **Real-time fluid dynamics for games**
  J. Stam, *Proceedings of the game developer conference*, 2003
- **Navier-Stokes Equations**
  T. Crawford (University of Oxford), *Numberphile*, 2019
  youtube.com/watch?v=ERBVFcutl3M
- **Divergence and Curl**
  G. Sanderson, *3Blue1Brown*, 2018
  youtube.com/watch?v=rB83DpBJQsE




Hochschule Niederrhein
University of Applied Sciences
Elektrotechnik und Informatik
Faculty of Electrical Engineering and Computer Science

# Quellcode und Demo

- Teil einer eigenen OpenGL-Engine
- Veröffentlicht unter `github.com/Zang3th/SalinityGL`
- MIT-Lizenz






Hochschule Niederrhein
University of Applied Sciences
Elektrotechnik und Informatik
Faculty of Electrical Engineering and Computer Science

Vielen Dank für Ihre Aufmerksamkeit!



Hochschule Niederrhein
University of Applied Sciences
Elektrotechnik und Informatik
Faculty of Electrical Engineering and Computer Science