

Strömungen in komplexen Geometrien

Felix Spanier, Rainer Grauer

Institut für Theoretische Physik I, Ruhruniversität Bochum



Einleitung

Thema dieser Arbeit ist die Simulation von inkompressiblen Strömungen in verschiedenen Geometrien, sowohl 2- als auch 3-dimensional. Für die Simulation der Navier-Stokes-Gleichung (NS)

$$\partial_t \vec{u} = -(\vec{u} \cdot \nabla) \vec{u} - \nabla p + \nu \Delta \vec{u} \quad (1)$$

$$\nabla \cdot \vec{u} = 0 \quad (2)$$

werden Non-Oscillatory-Schemes verwendet, die eine gute Auflösung von Schocks versprechen. Es wird auch gezeigt, daß Energie und Enstrophie gut erhalten sind.

Da die NS unter der Randbedingung der Inkompressibilität gelöst werden, benötigt man Verfahren, die die Divergenzfreiheit der Geschwindigkeit sicherstellen. Hierzu werden drei Methoden vorgestellt.

Geometrien werden über eine Penalty-Methode realisiert, die auf einem orthogonalen Gitter basiert, auf dem die Randwerte der Geometrie interpoliert werden.

Methoden

Drei verschiedene Lösungsansätze:

- RK: Lösung der Gleichungen für Stromfunktion und Vortizität. Direkt divergenzfrei durch Rotationsbildung

$$\Delta \Psi = -\omega \quad (3)$$

$$\vec{u} = \nabla \times \Psi \quad (4)$$

$$\partial_t \omega = -\vec{u} \cdot \nabla \omega + \nu \Delta \omega \quad (5)$$

$$\Delta \Psi = -\omega \quad (6)$$

- PM2: Direkte Lösung der Navier-Stokes-Gleichung, Berechnung eines Korrekturfeldes Φ die Divergenz, daraus addiert sich der Druck

$$\frac{\vec{u}^* - \vec{u}^n}{\Delta t} + \nabla p^{n+1/2} = -[(\vec{u} \cdot \nabla) \vec{u}]^{n+1/2} + \frac{\nu}{2} \nabla^2 (\vec{u}^n + \vec{u}^*) \quad (7)$$

$$\Delta t \nabla^2 \phi^{n+1} = \nabla \cdot \vec{u}^* \quad (8)$$

$$\nabla p^{n+1/2} = \nabla p^{n-1/2} + \nabla \phi^{n+1} - \frac{\nu t}{2} \nabla \nabla^2 \phi^{n+1} \quad (9)$$

$$\partial_t \vec{u}^n = -((\vec{u} \cdot \nabla) \vec{u})^{n+1/2} + \nu \Delta \vec{u}^n = \vec{u}^{n+1} \quad (10)$$

PM3: Variation von PM2, bei dem der Druck direkt berechnet wird

$$\Delta p = \nabla \cdot \vec{n} + \frac{\nabla \cdot \vec{u}^n}{\Delta t} \quad (11)$$

$$\vec{u}^{n+1} = \vec{u}^n + \Delta t (\vec{n} - \nabla p) \quad (12)$$

- Mix: Geschwindigkeit wird aus Ψ berechnet, damit die NS gelöst, daraus dann ω berechnet (s. a. GL. 5)

$$\vec{u}^n = \nabla \times \Psi^n \quad (13)$$

$$\partial_t \vec{u} = \dots \quad (14)$$

$$\partial_t \omega = \nabla \times (\partial_t \vec{u}) \quad (15)$$

$$\omega^{n+1} = \omega^n + dt \cdot \Delta \omega \quad (16)$$

$$\Delta \Psi^{n+1} = -\omega^{n+1} \quad (17)$$

Für die spätere Verwendung mit komplexen Geometrien kommen nur PM2/3 und Mix in Frage, da hier direkt in den Geschwindigkeiten gerechnet wird.

CWENO

CWENO ist ein Essential-Non-Oscillatory-Scheme, mit 2. Ordnung Genauigkeit im Ort und guter Auflösung von Schocks, mit Gewichtung.

- Das Verfahren ist flußerhaltend
- In den Zellen werden Parabeln rekonstruiert, am Rand der Zellen bilden sich Schocks aus, die weitertransportiert werden.

- Die Parabeln werden so gewählt, daß im glatten Fall die maximale Genauigkeit, bei Schocks aber die beste Schockauflösung erreicht wird.
- Vergleich zwischen optimaler Schockauflösung (non-limiting) und optimaler Genauigkeit (limiting).

Energie und Enstrophie

- Betrachtung der Erhaltungsgrößen Energie (E) und Enstrophie (Ω)

$$E = \int |u|^2 dV \quad (18)$$

$$\Omega = \int |\omega|^2 dV \quad (19)$$

- Anfangsbedingungen: zwei gleichsinnige Gauss-Vortizes mit $u_{max}=0.27$ und $\nu=2 \cdot 10^{-6}$
- Numerische Dissipation führt zu verstärkten Verlusten.

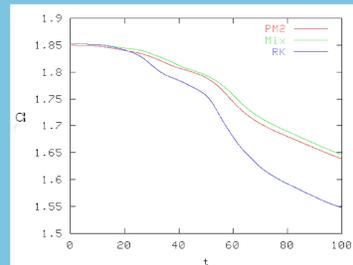


Fig. 1: Enstrophie

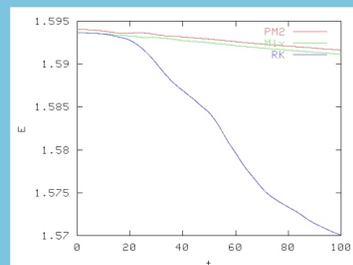


Fig. 2: Energie

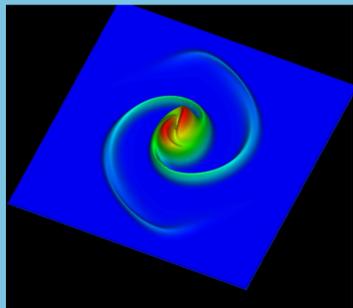


Fig. 5: ω zur Zeit $t=0$

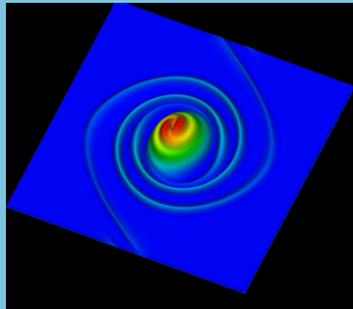


Fig. 6: ω im RK-Verfahren a) $t=50s$ b) $t=90s$

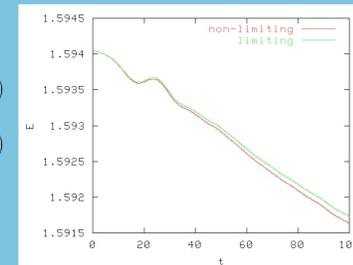


Fig. 3: Energie für PM2

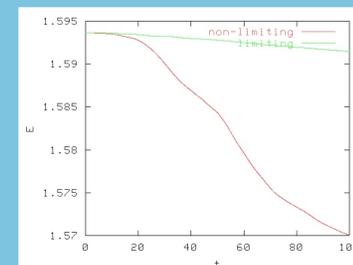


Fig. 4: Energie für RK

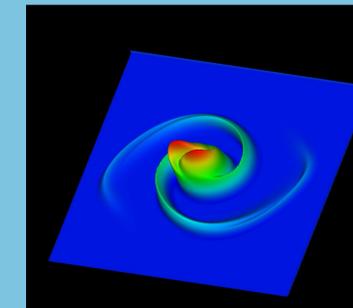


Fig. 7: ω im PM-Verfahren a) $t=50s$ b) $t=90s$

Penalty-Methoden

- Vorgabe einer Geschwindigkeit auf dem Rand der Geometrie
- Berechnung der Kraft aus der Differenz zwischen Soll- und Ist Geschwindigkeit (Gl. 23)
- Da nicht alle Punkte der Geometrie auf dem Gitter liegen (Abb. 8 c), wird die Sollgeschwindigkeit auf dem nächsten Gitterpunkt (Abb. 9: grüne Punkte) interpoliert durch Sollgeschwindigkeit und Interpolationspunkt (rote Punkte)

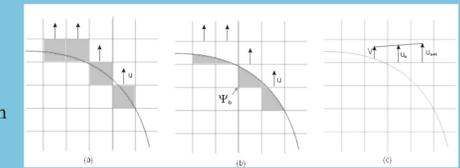


Abb. 8: a) keine Interpolation b) Volume Fraction c) lineare Interpolation

- Zusätzlich wird am Rand des berechneten Gebietes ein konstanter Einstrom erzeugt, der von-Neumann-Randbedingungen für den Druck erfordert

$$\frac{\partial p}{\partial x} = -\partial_t u_x - u_x \partial_x u_x - u_x \partial_y u_y + \nu \Delta u_x \quad (22)$$

$$\frac{\partial p}{\partial x} = \frac{2\nu}{(\Delta x)^2} u_x(1, j) \quad (23)$$

- Der Algorithmus für die Penalty-Methode sucht von einem gegebenen Punkt auf der Geometrie immer den nächstliegenden Gitterpunkt zur Geometrie, der außerhalb liegt
- Im Inneren kann die Geometrie sowohl als Hohl- wie auch als Vollkörper behandelt werden, ohne Änderung der äußeren Strömung

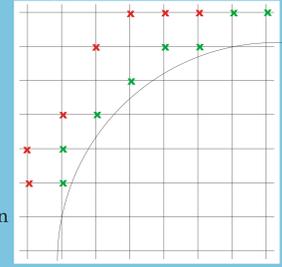


Abb. 9: Gitterpunkte, auf die eine Kraft wirkt, und Interpolationspunkte

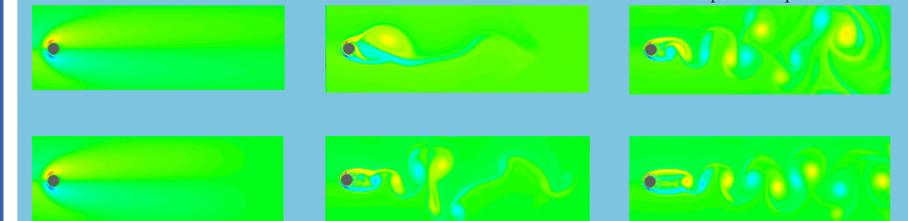


Fig. 10-12: Umströmte Kugel mit (v.l.n.r) $Re=7.5, 600, 30,000$ zu $t=50s$ (oben) und $t=100s$ (unten)

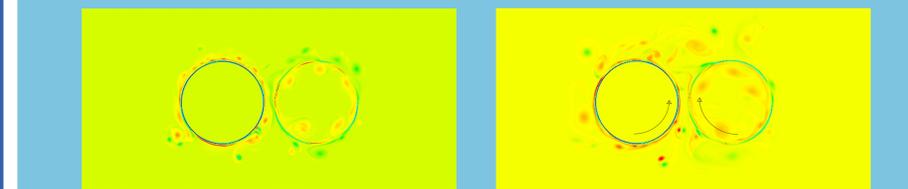


Fig. 13: Rotierende Walzen mit $u=0.4$ und $Re=28,000$ a) $t=50$ b) $t=100$

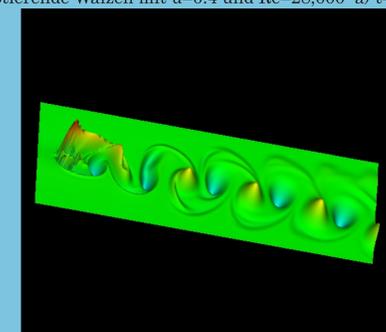


Fig. 14: Umströmter Zylinder $Re=600$ und $t=287s$

Legende

\vec{u}^n	Geschwindigkeit bei $t = n \cdot \Delta t$
\vec{u}_b	Sollgeschwindigkeit am Rand
$\omega = \nabla \times \vec{u}$	Vortizität
$\Psi = \Delta \omega$	Stromfunktion
ν	Viskosität
\vec{n}	Nichtlinearität
p	Druck
ϕ	Korrekturfeld

Ergebnisse und Ausblick

- Geschwindigkeitsbasierte Methoden (Mix, PM2) liefern mit CWENO deutlich geringere numerische Dissipation
- Das Mix-Verfahren läßt auch inkompressible Simulationen mit AMR zu
- Penalty-Methoden liefern für einen weiten Bereich von Reynolds-Zahlen stabile Methoden zur Simulation komplexer Geometrien
- Penalty-Methoden in Mix erlauben auch AMR-Simulationen komplexer Geometrien
- Ausblick
- 3dimensionale Simulationen komplexer Geometrien (in Arbeit)

Literatur

- [1] Kurganov, A. & Levy, D. [2000] *SIAM J. Sci. Computing* **22** 4
- [2] Brown, D.L., Cortez, R. & Minion, M.L. [2001] *J. of Computational Physics* **168**
- [3] Fadlun, E.A., Verzicco, R., Orlandi, P. & Mohd-Yusof, J. [2000] *J. of Computational Physics* **161**
- [4] H.E. Johnston & J.G. Liu [2002] *Journal of Computational Physics*