



Digitales Sommersemester 2021:

# Datenvisualisierung und GPU-Computing

## Übung 5: Wahlaufgabe

16.06.2021

Andreas Beckert

[andreas.beckert@uni-hamburg.de](mailto:andreas.beckert@uni-hamburg.de)

*Regionales Rechenzentrum, Visual Data Analysis Group,  
FB Informatik Arbeitsgebiet „Scientific Visualization and Parallel Processing“*

# Diese Woche

- Besprechung von Übung 4 (Streamlines)
- Übung 5:
  - Hinweis: Essay / Klausurersatz
  - Wahlaufgabe
  - Abgabe: Mo. 26. Juli 2021
- Nächste Übung: 30. Juni 2021

# Übung 4: Stromlinien

Mit dem derzeitigen Stand unseres Visualisierungsprogramms können wir zwar die einzelnen Komponenten der erzeugten „Wind“-Vektoren als Skalarfelder betrachten, das 3D-Strömungsfeld des „synthetischen Tornados“ an sich lässt sich damit jedoch schwer analysieren. Als nächstes wollen wir daher dieses Strömungsfeld visualisieren. In der Vorlesung wurden u.a. Stromlinien (Streamlines) als eine Möglichkeit besprochen, die Windrichtung an verschiedenen Stellen des 3D Volumens darzustellen.

## Aufgabe 4:

- Implementiere Stromlinien in das Visualisierungsprogramm und erkunde damit das Strömungsfeld des synthetischen Tornados.
- Stelle die Stromlinien (zunächst) als einfache Liniengeometrie dar, analog zu den Konturlinien aus Übung 3 (also ohne Beleuchtung).
- Vergleiche Integration mit dem Euler- und dem Runge-Kutta-Verfahren.

# Übung 4: Leitfragen und Tipps

- Als „Template“ für den `StreamLinesMapper` und `StreamLinesRenderer` für die Stromlinien bieten sich die beiden Klassen aus Übung 3 an.
  - Wer möchte: Der Renderer unterscheidet sich kaum (es macht sicher Sinn, eine andere Farbe zu verwenden), daher wäre an dieser Stelle eine Möglichkeit über Vererbung die Struktur des Programmes zu vereinfachen.
- Es bietet sich wieder an, zunächst im Mapper ein beliebiges Liniensegment zu erzeugen und damit zu prüfen, dass der Renderer funktioniert.
- Bei der Entwicklung bietet es sich an, den Horizontalschnitt aus Übungen 2 und 3 als Hilfe zu verwenden: Stimmt die Richtung der Stromlinien mit dem Vorzeichen der betrachteten Windkomponente überein?
- Wie sollte das Programm den Fall behandeln, dass eine Stromlinie bei der Berechnung das Datenvolumen verlässt?

# Übung 4: StreamLinesMapper

- In dieser Klasse benötigen wir wieder eine Methode `setDataSource(...)`, diese kann aus Übung 2 bzw. 3 übernommen werden.
- Hier bietet es sich an, für die unterschiedlichen Integrationsverfahren jeweils eine Methode zu implementieren z.B. `EulerIntegration(QVector3D position, float stepSize)`. Diese Funktion soll dann jeweils die Endposition eines Liniensegmentes einer Stromlinie zurückgeben.
- In der Methode `computeStreamLines(QVector<QVector3D> seedPoints)` werden dann für jeden „Seed Point“ die Stromlinien berechnet. Diese wird dann von der Methode `updateStreamLines()` der Klasse `StreamLineRenderer` aufgerufen und gibt die Liniensegmente zurück.
- Zur Berechnung der Stromlinien benötigen wir noch eine weitere Methode, `interpolateWindVector(QVector position)` die den interpolierten Wert des Windvektors zurückgibt.

# Übung 4: StreamLinesMapper / StreamLinesRenderer

- Tipp: Es ist sinnvoll für die Integration den “World Space” (0...1) zu verwenden. Lediglich für die Interpolation ist es notwendig, die Koordinaten des “World Space” in die Koordinaten unseres Gitters (0...nGridPoints-1) zu transformieren.
- Es bietet sich an in der Methode `updateStreamLines()` der Klasse `StreamLinesRenderer` die Positionen der “Seed Points” der Stromlinien zu erstellen und anschließend die “Seed Points” an die Methode `computeStreamLines(...)` vom `StreamLinesMapper` zu übergeben.

# Übung 4: Zielfragen

- Sobald die Berechnung funktioniert, betrachte Stromlinien, die am Punkt  $(x,y) = (0.2, 0.2)$  sowie in regelmäßigen Abständen von 0.1 entlang der z-Achse gestartet werden. Was kann über das Strömungsfeld ausgesagt werden?
- Für  $z = 0.8$  vergleiche Euler- und RK-Integration für eine Schrittweite von 0.2 und eine maximale Schrittzahl von 100. Wie unterscheiden sich die berechneten Stromlinien? Passt die Euler-Integration zur x-Komponente des Windfelds (betrachte mit dem Horizontalschnitt)?
- Starte nun Stromlinien auf einem regelmäßigen Gitter von 0.1 in allen drei Raumrichtungen. Ist die Visualisierung noch lesbar?
- Verringere den Abstand der Seed Points auf 0.01 (eine Dimension nach der anderen...). Dauert die Berechnung schon länger?

# Hinweis: Essay / Klausurersatz

Da wir dieses Semester keine Prüfungen durchführen dürfen, werden wir die ursprünglich geplante Beschreibung der Übungsaufgaben erweitern und als eine Art Essay benoten.

Im Essay sollen eine Reihe von Leitfragen beantwortet werden, die u.a. auch die Übung in den Kontext der Vorlesung stellen. Vordergründig sollt ihr damit zeigen, dass ihr euch mit dem Vorlesungs- sowie Übungstoff auseinandergesetzt und die Konzepte der Vorlesung verstanden habt.

Die Abgabe wird etwas umfangreicher als die ursprünglich geplanten 5 Seiten, da diese auch als Klausurersatz gilt.

- Deadline: 26.07.2021, Abgabe vom Essay sowie vom Code
- Es wird keine Zweitprüfung geben!
- Details über Länge der Abgabe und Leitfragen werden in der letzten Vorlesungswoche bekanntgegeben.
- Der Code der Übung wird dabei nicht benotet, allerdings ist es wichtig, dass die schriftlichen Ausarbeitungen individuell sind.



# Übung 5: Wahlaufgabe

Wir haben verschiedene Themen der Visualisierung in der Vorlesung und bisherigen Übung besprochen. Wählt ein Thema, welches euch interessiert, und ergänzt euer Programm in diese Richtung. Es muss keine umfangreiche Implementierung sein, aber **das Thema soll in der schriftlichen Abgabe / Essay** (Klausurersatz) dokumentiert sein.

Beispiele können sein:

- Animiert die Visualisierung über die Zeit.
- Färbt die Stromlinien nach „Wichtigkeitskriterien“ wie der Krümmung der Linien ein.
- Implementiert eine wahrnehmungslineare Colourmap und vergleicht die Colourmap aus Übung 2 damit.
- Parallelisiert das Programm mit OpenMP oder programmiert die Berechnungen der Stromlinien in den Shader

Oder untersucht eine andere Datenquelle mit den Verfahren, oder beleuchten die Stromlinien, oder ...

# Abgabe der Wahlaufgabe

- Eine Bearbeitung der Programmieraufgaben der Übung ist Voraussetzung, um die Wahlaufgabe zu bearbeiten sowie das Essay zu schreiben. Als Nachweis reicht ihr zum Semesterende **euren Quellcode als ZIP-Archiv** sowie das Essay (Klausurersatz) ein. Die genauen Leitfragen für das Essay werden noch in der Vorlesung bekanntgegeben.
- Auf die Umsetzung der Wahlaufgabe soll ebenfalls im Essay eingegangen werden.
  - Beschreibung und Diskussion der Wahlaufgabe
  - Screenshots (Ergebnisse der Aufgaben bzw. eurer Wahlaufgabe, ggf. auch mit Problemen)
  - Alle weiteren Leitfragen zum Essay werden in der letzten Vorlesungswoche bekanntgegeben.
- Deadline: Montag **26. Juli 2020**