

**FELLOWSHIP FÜR INNOVATIONEN
IN DER DIGITALEN HOCHSCHULLEHRE**

Hochschule Ostwestfalen-Lippe

Fachbereich 6 Maschinentechnik und Mechatronik

Liebigstr. 87, 32657 Lemgo

05261 - 702 5773

georg.klepp@hs-owl.de

Virtual Fluid Experience

Abschlussbericht Virtual Fluid Experience

Lehrinnovation 2018

Kurzfassung

Lehrinnovation

Eine Reihe von Applikationen zur Veranschaulichung von Grundlagen der Fluidmechanik wurden erstellt. Die dazu benutzten Beispiele sind alltagsnah und praxisrelevant. Die Eingabemöglichkeiten durch den Nutzer sind auf die wichtigsten Parameter in einem sinnvollen Größenbereich reduziert. Die Darstellung der Ergebnisse folgt weitgehend der Bildsprache der Computational Fluid Dynamics (CFD). Die Zugänglichkeit der Applikationen auf unterschiedlichen Plattformen ist gewährleistet, durch die Verwendung von Matlab als Basis. Die Applikationen können zur Vorbereitung sowie Nachbereitung des Inhalts der Präsenzveranstaltungen verwendet werden.

Erreichte Ziele und “lessons learnt”

Die Lehrinhalte wurden wie geplant in geeignete Aufgaben implementiert und in Applikationen umgesetzt. Die Lehrinnovation wurde mit ausgewählten Studenten getestet, aber noch nicht über die volle Länge einer Lehrveranstaltungen eingebracht. Dies soll im Sommersemester 2019 geschehen und wird geeignet begleitet.

Ob der für die Zielgruppe didaktisch passende Ansatz identifiziert wurde oder in wie weit noch eine Anpassung nötig ist, kann erst die Erfahrung nach mindestens einem Durchgang zeigen. Um die Akzeptanz und Nutzung neuer unbekannt Formate zu erreichen ist hier neben der ausführlichen Vorstellung des Formats auch ein entsprechender “Pull” nötig (Hörsaalübung, Hausaufgaben).

Das größte Problem in Umsetzung, Verbreitung und Verstetigung ist die Wahl der geeigneten technischen Plattform zur Realisierung. Ein Problem liegt in einer effizienten Umsetzung der rechenintensiven Anteile, ein zweites in der Verbreitung, die möglichst plattformunabhängig sein soll, sowohl jetzt als auch in der Zukunft.

Um ein nahtlose Darstellung zu gewährleisten, müssen die im Hintergrund laufenden numerischen Algorithmen in Sekundenschnelle eine Lösung berechnen. Für einige der umgesetzten Aufgaben sind dazu besonders leistungsfähige Algorithmen (zur Lösung großer nicht-linearer Gleichungssysteme) nötig. Dies führt zu Einschränkungen in der Wahl der Programmiersprache sowie zu hohen Ansprüche bezüglich der Qualität der verwendeten Algorithmen.

Eine weitere sehr große Herausforderung ist die aktuell und zukünftig plattformübergreifende graphische Darstellung der Ergebnisse und ggf. die Eingabe über eine graphische Nutzeroberfläche (GUI). Im Projekt sollte hier möglichst auf bestehende Lösungen zurückgegriffen werden. Eine Möglichkeit dazu wäre Java. Diese plattform-unabhängige Sprache war in der Vergangenheit auch für Lern-Apps sehr beliebt. Aktuell ist diese jedoch nur noch stark eingeschränkt verwendbar: aus Sicherheitsgründen wird die Installation vielfach nicht mehr unterstützt. Viele der Applikationen können nicht mehr einfach hochgeladen werden und werden anscheinend auch nicht mehr gepflegt.

Aus diesen Gründen wurden die Applikationen schliesslich in Matlab implementiert. Dadurch ist einerseits die Qualität der numerischen Algorithmen gewährleistet. Andererseits sind durch die Verwendung dieses Formats die Applikationen in dem weit verbreiteten und zunächst auch weiter gepflegten “Ökosystem” Matlab verankert. Vor allem im akademischen Umfeld ist Matlab weit verbreitet. Durch die Kompatibilität mit der freien Software Octave ist

durch entsprechende Programmierung auch eine Nutzung möglich, wenn keine Matlab-Lizenz vorliegt.

Verstetigung

Die Applikationen werden zunächst an der Hochschule Ostwestfalen-Lippe im Fach Fluidodynamik verwendet. Nach einem ersten erfolgreichen Durchlauf ist geplant die Applikationen über Matlab-File Exchange weiter zu verbreiten. Die Pflege der dann bestehenden Skripte würde zunächst vom Labor für Strömungsmaschinen und Fluidodynamik übernommen werden, im Rahmen der Pflege der Unterlagen für die Lehre. Dadurch ist eine Verstetigung und Verbreitung der Lehrinnovation gewährleistet.

Übertragbarkeit

Der gewählte Ansatz kann auch auf andere Lehrinhalte, die ähnlich in den Ansprüchen sind, übertragen werden. Möglich wären hier Thermodynamik, Wärmetransport, Festkörpermechanik, Energietechnik. Im Bereich der Energie und Wärmeübertragung gibt es mit Energy2D auch vergleichbare Ansätze, die jedoch wegen der Einschränkungen bei der Anwendung von Java, mittlerweile in Gebrauch und Verbreitung eingeschränkt sind. Neben der Verwendung als "Hausaufgabe" zur Einarbeitung in neue Lehrinhalte, können die Apps auch in einer Lehrveranstaltung, als virtuelles ad-hoc Experiment, eingebaut werden.

Ziele

Das Ziel des Projektes ist es, den Studierenden im Fach Fluidmechanik eine virtuelle Erfahrung zu ermöglichen, die den Zugang und das Verständnis für die theoretischen Zusammenhänge der Fluidmechanik erleichtert. Dadurch sollen die Noten und die Durchfallquoten im Fach Fluidmechanik gesenkt werden. Dies soll es auch ermöglichen, für die Studierenden bessere Ergebnisse in Regelstudienzeit zu erzielen.

Ein weiteres Ziel ist es, Erfahrungen mit dem Einbinden von e-learning Elementen in der Lehre zu sammeln. Hier ist e-learning nicht mehr eine Alternative zu oder eine von mehreren Komponenten der traditionellen Lehre, sondern sie wird wesentlicher Bestandteil der Wissensvermittlung auch in Präsenzveranstaltungen.

Konzept Virtual Fluid Experience

Eine Lern-App wurde entwickelt, die es den Studierenden ermöglicht, verschiedene Lerninhalte der Fluidmechanik in einer virtuellen Lernumgebung zu simulieren, und so virtuelle Experimente zum Thema durchzuführen. Durch die Nutzung einer virtuellen Umgebung, die über vielfältige digitaler Formate zugänglich ist, werden die Lebenswelten junger Menschen in die Lehre und Wissensvermittlung integriert.

Das Lernprinzip, welches hinter Virtual Fluid Experience steht, ist handlungsorientiert. Die Studierenden sind aktiv und konstruktiv tätig und sammeln nach dem „trial-and-error“ Prinzip eigene Erfahrungen. Diese handlungsorientierte Komponente ist im bisherigen Unterricht wenig vertreten.

Der Wissenserwerb erfolgt in den einzelnen Lernphasen in einem vom Studierenden aktiv selbst gesteuerten Prozess nach dessen eigener Vorprägung und den eigenen Bedürfnissen. Dieser Prozess wird von Seiten des Lehrenden lediglich unterstützt und begleitet. Durch den Bezug zu konkreten Anwendungen aus dem Alltag sowie der beruflichen Praxis wird auf Seiten der Studierenden das Interesse geweckt und ein Bezug geschaffen

Dadurch wird vor allem den Studierenden mit Problemen bei abstrakten Fragestellungen der Einstieg erleichtert. Darum sollen vor allem die grundlegenden Begriffe und Zusammenhänge vermittelt werden. Auf diese Weise wird der Anteil an nicht ausreichenden Prüfungsgleistungen verringert.

Erfahrungen aus ähnlichen Projekten

Eine Auseinandersetzung mit Veröffentlichungen zu diesem Thema zeigt, dass die Verwendung von Simulation in der Lehre von Fluidmechanik zu positiven Ergebnisse führt. Das Interesse der Studierenden der Ingenieurwissenschaften an Computer-zentrierten Themen ist groß und durch die Verwendung zusätzlicher Lehrmethoden wird das Interesse gesteigert. Wichtig ist die Einbindung ins Curriculum, d.h. die gelungene Abstimmung mit Theorie und Versuch, die parallel dazu vermittelt werden, sowie die eigenständige Bearbeitung von Aufgaben oder Projekten. Meist werden diese Methoden bei

fortgeschrittenen Studenten eingesetzt, die mit den Grundlagen schon vertraut sind, im vorliegenden Projekt ist dies aber nicht möglich.

Ein zusätzliches Spannungsfeld zeichnet sich durch die Abstimmung der gewählten Applikation mit den Kompetenzen der Studierenden ab: Ist die Komplexität des verwendeten Werkzeugs zu groß wird ein zu großer Teil des Aufwandes auf das Erlernen des Werkzeugs verwendet, ist die Anwendung / das Beispiel zu trivial wird das Tool als Spielzeug benutzt und der Lehrinhalt geht verloren.

Es wurde gezeigt, dass durch die Verwendung von Tools, die auf die Bedürfnisse des Curriculums angepasst sind, und durch spezielle voreingestellte Aufgaben, die dann frei bearbeitet werden können, gute Lernergebnisse erzielt werden können. Die Bandbreite der dazu verwendeten Werkzeuge reicht von kommerziellen Programmen mit vorgegebenen Voreinstellungen bis hin zu massgeschneiderten Eigenentwicklungen. Ein grosses Problem scheint die Verfügbarkeit und Verbreitung der beschriebenen Lehrinnovationen zu sein: nach dem Ende der entsprechenden Projekte wird nur noch ein Teil weiter gepflegt.

Ähnliche Werkzeuge/Applikationen

Ansätze Simulationswerkzeuge in der Lehre der Strömungstechnik und Fluidmechanik zu verwenden gibt es verstärkt schon seit 20 Jahren. Das sind einerseits speziell entwickelte didaktische Anschauungsbeispiele. Das andere Extrem sind voll funktionsfähige Simulationswerkzeuge (CFD) mit deren Hilfe Beispiele gerechnet werden können. Durch die stetige Innovation im IT Bereich ist die mittel- und langfristige Pflege solcher Werkzeuge aufwändig, so dass viele dieser Anwendungen nach Beendigung des Projektes ihrer Erstellung anscheinend nur eine kurze Lebensdauer haben. Im folgenden eine (unvollständige) Liste, und deren Einordnung in Bezug auf die Applikationen der Virtual Fluid Experience:

Online Experimente in der Strömungslehre, TU Berlin (owl.hermann-foettinger.de) : Applikationen in Java, gleiches Ziel, ähnliche Herangehensweise aber signifikant höheres Niveau der gewählten Beispiele. Letztes Update 2011, lässt sich wegen schlechter Java-Kompatibilität leider kaum noch nutzen.

FlowLab (Fluent), CFDStudio, CFDnet, VirtualFlowLab, chimeraCFD,...: Computational Fluid Dynamics (CFD) mit maximal möglichen Voreinstellungen und minimalen Nutzereingabe, speziell zum Einsatz in der Lehre, kaum / nicht mehr gepflegt. Liesse sich durch zusätzliche Voreinstellungen (v.a. numerischer Parameter) anpassen, Voraussetzung sind leistungsfähige Gleichungslöser. Bei einigen Web-basierten Lösungen Probleme mit Browserkompatibilität.

Flowsquare, ... : 2D CFD-Löser für Windows (s.a. oben). Es gibt eine Reihe ähnlicher Löser, die noch gepflegt werden. Diese sind meist jedoch nicht auf allen gewünschten Plattformen lauffähig. Ist der Programmcode verfügbar, können entsprechende Applikationen erstellt werden, indem bestimmte Parameter festgelegt und dem Nutzerzugriff entzogen werden.

CFDCalc (Altair) : Massgeschneiderte Nutzeroberfläche für konkrete Aufgabe, Löser läuft im Hintergrund. Hervorragend als Plattform geeignet, jedoch Freigabe zur Nutzung von Altair verweigert

Energy2D (Concord) [13] : Simulationsumgebung in Java speziell für die Lehre. Fokus auf Wärmetransport und Energietechnik. Gut geeignet, muss aber den Bedürfnissen der strömungstechnischen Fragestellungen entsprechend umgeschrieben/erweitert werden. Transfer auf Javascript würde aktuelle Kompatibilitätsprobleme lösen, angedacht aber nicht terminiert.

Matlab [14] : Matlab Applikationen zur Lehre in Strömung und Wärmetransport. Gewählte Beispiele anspruchsvoller und komplexer als Beispiele für VirtualFluid Experience. Als Blaupause für VirtualFluidiExperience geeignet.

Festlegen der Inhalte

Zunächst wurde eine Analyse der Rückmeldung zu den Übungsaufgaben, der Selbsttests sowie der Klausuren durchgeführt. Daraus wurden folgende Themen abgeleitet, die durch passende Beispiele dargestellt werden:

- Inkompressible Fluide. Eindimensionale Strömung: Zusammenhang zwischen: Geschwindigkeit (Querschnittsänderung), Druck, Höhe, Einfluss der Dissipation. Einflussgrößen schrittweis in dieser Reihenfolge in verschiedene Apps/Aufgaben implementiert. Stoffwerte für Luft, Wasser, Öl. Variante Hydrostatik mit Druck, Höhe, Stoffwerte. Darstellung von Geschwindigkeit und Druck.
- Umströmung eines Körpers (Zylinder bzw. Ellipse). Parameter: Geometrie (Durchmesser und Winkel), Stoffwerte (Luft, Wasser), Anströmgeschwindigkeit / Reynoldszahl. Darstellung der Druckverteilung und der Stromlinie.
- Kompressible eindimensionale isentrope Strömung: Zusammenhang zwischen Zustandsgrößen, Geschwindigkeit und Querschnittsänderung, Stoffwerte für Luft. Darstellung Geschwindigkeit und Zustandsgrößen.

Die Beispielaufgaben sollten realitätsnah sein, den fachlichen Inhalt anschaulich vermitteln sowie effektiv und effizient umsetzbar sein. Im Gespräch mit Studierenden wurde im Vorfeld Feedback zur Akzeptanz eingeholt.

Festlegen der Umsetzung

Für die genannten Aufgaben werden die ein- und zweidimensionalen Erhaltungsgleichungen gelöst. Das in der Aufgabenstellung definierte Berechnungsgebiet wird geeignet diskretisiert und für diese Punkte die gesuchten Variablen (Druck und Geschwindigkeiten) berechnet. Dazu werden dann grosse lineare Gleichungssysteme iterativ gelöst. Der grösste Teil der Stoffwerte und Randbedingungen ist durch die Aufgabenstellung vorher festgelegt. Im Rahmen festgelegter Grenzen kann der Nutzer nur einige wenige Parameter frei wählen.

Die Ergebnisse werden graphisch anschaulich dargestellt, wichtig ist hier der sofort sichtbare qualitative Vergleich unterschiedlicher Strömungsformen, weniger das Extrahieren konkreter quantitativer numerischer Werte.

Festlegen der Plattformen

Ziel ist es die Applikationen auf möglichst vielen Plattformen zugänglich zu machen sowie so weit wie möglich unabhängig von Hardware und Software zu gestalten

Folgende Plattformen sind möglich .

PC : Betriebssysteme Windows, Linux, Mac

Mobil : Betriebssystem Android, iOS

Cloud : Installation auf Server, Zugang über Webbrowser

Browser : lokale Ressourcen, Nutzerinteraktion über Webbrowser

Eine Möglichkeit ist, als Basis einen Algorithmus zur Lösung der zweidimensionalen Navier-Stokes Gleichungen zu benutzen und mit einer passenden Benutzeroberfläche auszustatten. Dieser Ansatz ist sehr aufwändig in der Realisierung und in der Pflege und würde den Rahmen dieses Projektes sprengen. Fokus dieses Projektes soll die Umsetzung geeigneter Beispiele sein und nicht das Erschaffen einer (weiteren) Umgebung, um solche Beispiele erstellen zu können. Aus diesem Grund wurde zunächst nach einer passenden Software Plattform gesucht, um die Beispiele möglichst effizient und effektiv umsetzen zu können.

Zunächst wurde der Ansatz verfolgt, die Anwendungen auf der Basis von Energy2D zu gestalten. Dazu wurden auf der Basis von Energy2D Anwendungen zu den oben genannten Themen erstellt. Der Testlauf mit diesen Anwendungen zeigte aber folgende Probleme auf, die sich aus der Programmierung in Java sowie dem Schwerpunkt Wärmetransport ergaben:

- Probleme in der Kompatibilität. Applikationen können nur eingeschränkt aufgerufen werden. Vorab-Installation von Programmkomponenten nötig, die manchmal nicht reibungsglos läuft. Kein Zugang über Browser möglich.
- Graphische Darstellung der berechneten Felder und Randbedingungen, die vorhanden sind, sind nur bedingt auf Strömungsfelder sinnvoll übertragbar.
- Erweiterung der gelösten Gleichungen nötig, um Druckkomponente der Strömung zu berechnen.

Ein Neuprogrammieren in z.B. Javascript würde den Rahmen dieses Projektes sprengen, ein aufwändiges Umprogrammieren in einer wenig unterstützten Sprache nicht zukunftsweisend. Aus diesem Grund wurde dann die Arbeit mit Energy2D eingestellt und eine Alternative gesucht.

Die Wahl für eine geeignet Softwareumgebung zur Gestaltung der Anwendungsbeispiele fiel dann auf Matlab: Hier sind geeignete Algorithmen für eine schnelle numerische Berechnung von Druck- und Geschwindigkeitsfeldern sowie deren passende graphische Darstellung gegeben.

Matlab sowie die Freeware Variante Octave sind vor allem im akademischen Umfeld sehr beliebte Werkzeuge. Matlab-Skripte sind verbreitet sowie auch mit anderen als der aktuellen

Matlab-Version kompatibel. Die Probleme der Distribution sowie der (softwareseitigen) Verstetigung wären damit gelöst.

Matlab ist für Windows, Linux und Apple verfügbar. Für diese Betriebssysteme können auch stand-alone Apps generiert werden, die keine Matlab-Lizenz benötigen. Von Matlab gibt es auch eine mobile Applikation sowie eine Cloud-Applikation in der entsprechende Skripte laufen können. Damit ist die gewünschte Plattform-Vielfalt gewährleistet.

Arbeitsplan

Projektstart war der 1. März 2018. Aktuell werden die erstellten Applikationen und Beispiele getestet und ggf. geeignet nachjustiert. Zwischenstände zum Erfolg der Umsetzung erwarten wir im Sommer 2019 sowie im Sommer 2020.

Im Folgenden ein kurzer Überblick über die Arbeitspakete:

Arbeitspaket Konzeptionierung (12 Wochen)

Es wurden Studenten befragt, ein Konzept mit den Themen erstellt sowie geeignet Anwendungsbeispiele definiert. Daraus wurde dann ein Satz konkreter Aufgaben mit Geometrien, Stoffwerten und Randbedingungen abgeleitet. Die Abstimmung erwies sich aufwändiger als erwartet. Durchführung der Arbeiten durch wissenschaftliche und studentische Hilfskräfte.

Arbeitspaket Methodik Umsetzung (20 Wochen)

Zunächst wurde versucht die Beispielaufgaben durch eine Weiterentwicklung auf der Basis von Energy2D umzusetzen. Nach einer kritischen Überprüfung wurde nach 6 Wochen abgebrochen. Ein neuer Ansatz wurde dann auf der Basis von Matlab gewählt. Dazu wurden, in Abhängigkeit von Themen und Aufgaben die unterschiedlichen Algorithmen definiert, die zur Verwendung kommen. Auch hier kann auf bekannte und vorhandene Algorithmen zugegriffen werden. Hilfreich erweist sich dabei die leistungsfähige mathematische Bibliothek. Durchführung der Arbeiten durch wissenschaftliche und studentische Hilfskräfte.

Arbeitspaket Interface (8 Wochen)

Definition der Schnittstelle Mensch-Maschine und deren Umsetzung, Zugänglichkeit auf unterschiedlichen technischen Plattformen. Hier wurde auf bereits vorhandene technische Lösungen (Energy2D, Matlab) zugegriffen, die nur noch angepasst werden mussten. Dadurch kann der Aufwand reduziert werden. Durchführung der Arbeiten durch wissenschaftliche und studentische Hilfskräfte.

Arbeitspaket Applikationen (10 Wochen)

Für die einzelnen Beispiele/Anwendungen sowie Plattformen wird die Software spezifisch angepasst. Die Applikationen werden implementiert und getestet. Durchführung der Arbeiten durch wissenschaftliche und studentische Hilfskräfte.

Arbeitspaket **Projektkoordination** (9 Wochen)

Organisatorische Betreuung und Dokumentation. Durchführung der Arbeiten durch wissenschaftliche und studentische Hilfskräfte.

Bei der Abarbeitung gab es zwei Probleme, die zu Verzögerungen führten:

- Der Wechsel der Plattform von Energy2D in Java auf Matlab und damit verbunden ein Neu-Start bei der Umsetzung.
- Finden geeigneter Mitarbeiter, die sowohl über ausreichend tiefe Kenntnisse der Fluidmechanik als auch der Programmierung verfügen, sowie Fluktuation der Mitarbeiter.

Literatur

[1] R. Felder, L. Silverman; Learning and Teaching Styles in Engineering Education. Engr. Education, 78(7) 1988

[2] J. Cimbala et al: Using Flowlab, a Computational Fluid Dynamics Tool, to Facilitate Teaching of Fluid Mechanics. Proc. IMECE04, 2004 ASME Int. Mech. Eng. Congress and Exposition Anaheim, IMECE2004-59870F , 2004

[3] C. Sert, G. Nakiboglu: Use of Computational Fluid Dynamics (CFD) in Teaching Fluid Mechanics. AC 2007-1560, American Society for Engineering Education, 2007

[4] F. Stern, H. Yoon: Hands on integrated CFD educational interface for introductory fluid mechanics. Int. J. Aerodynamics, Vol. 2, 2012

[5] J. R. Wilson, K. C. Gramoll: Viscous Fluid Dynamics App for Mobile Devices Using a Remote High Performance Cluster, 122nd ASEE Annual Conference and Exposition Seattle, paper ID #11877, 2015 .

[6] H. Hassan: An Interactive Fluid Dynamics Game on the iPhone. Master Thesis. Informatics Department, TU München, 2009

[7] D. Adair, M. Jaeger: Integration of Computational Fluid Dynamics into a Fluid Mechanics Curriculum. Computer Applications Engineering Education Vol. 22, 1, Wiley 2011

[8] K. Zbeeb, B. McDonald, S. Shin, P. Ravikumar: Introducing CFD Numerical Analysis in Fluid Dynamics to Junior Engineering Students. ASEE IL-IN Conference 2018, Lafayette

[9] F. Rückert et al: Computational Fluid Dynamics as a Driver for Teaching Millennials. Proceedings OOFHEC 2018, Aarhus

[10] E. Sauret, D. Hargreaves: Collaborative Learning Approach to Introduce Computational Fluid Dynamics. AAEE Conference 2015, Geelong

[11] F. Stern et al: Hands-on CFD-Interfaces for Engineering Courses and Laboratories. Journal Engineering Education, January 2006

[13] Energy2D: <http://energy.concord.org/energy2d/>

[14] Matlab:

<https://de.mathworks.com/videos/teaching-fluid-mechanics-and-heat-transfer-with-interactive-matlab-apps-81962.html>