

MAUS

Jörg J. Buchholz

Hochschule Bremen, Fachbereich Maschinenbau
Neustadtswall 30, D-28199 Bremen, Deutschland

Seit dem Wintersemester 2003/2004 wird Ingenieurmathematik im Studiengang ILST (Internationaler Studiengang Luftfahrtssystemtechnik und -management) an der Hochschule Bremen größtenteils als rechnerbasiertes Übungsseminar durchgeführt. Die Studierenden lösen, nach einer kurzen Einführung in die Thematik, in Anwesenheit des Dozenten in Kleingruppen am Rechner mit *Matlab* klassische Ingenieurmathematik-Aufgaben und beantworten schriftlich Fragen zu aufgetretenen Problemen und Besonderheiten bei der Lösung und zu Anwendbarkeitsbedingungen und Grenzen der Verfahren. Dieser Beitrag beschreibt erste Erfahrungen mit dieser Lehrveranstaltungsform und diskutiert mögliche Konsequenzen.

LEHRKONZEPT

Einführung

Spätestens seit der oft strapazierten Erkenntnis Lao Tses: *I hear and I forget, I see and I remember, I do and I understand*, weiß jeder Lehrende, dass wir Menschen nur das wirklich lernen, was wir selbst angefasst, umgesetzt, verwirklicht und immer wieder geübt haben. Die klassische Talk-and-Chalk-Vorlesung bringt daher in den seltensten Fällen mehr als das Durcharbeiten eines guten Lehrbuches; außer vielleicht der Tatsache, dass sich Studierende auf diese Weise eher genötigt fühlen, sich am Montag um 8 Uhr mit Mathematik zu beschäftigen, als durch ein Lehrbuch, das einsam auf dem Schreibtisch wartet.

Nach langjährigen ausgesprochen positiven Erfahrungen mit übungsbasiertem Lernen in den Modulen Regelungstechnik, Flugregelung und Systemsimulation hat sich der Fachbereich Maschinenbau [1] der Hochschule Bremen daher entschlossen, in seinem Internationalen Studiengang Luftfahrtssystemtechnik und -management [2] auch das Modul Ingenieurmathematik in dieser Lehrform als Mathematik - Übungs Seminar (MAUS) anzubieten. Momentan werden beispielsweise von den 8 Semesterwochenstunden Mathematik im ersten Semester nur 2 Semesterwochenstunden (die mit Einführung der Bachelor-Studiengänge ganz durch das Selbststudium von Lehrbüchern [3] ersetzt werden) als kurze Einführung in

die Thematik in Form einer Vorlesung genutzt; während der restlichen 6 SWS lösen die Studierenden am Rechner mit *Matlab* Übungsaufgaben und beantworten schriftlich Fragen zu aufgetretenen Problemen und Grenzen der Lösungsverfahren.

Kommunikation

Die dabei verwendete Kommunikationsstruktur ist in Abbildung 1 dargestellt.

In einer auf dem Server des Fachbereichs liegenden Datenbank befinden sich Aufgaben und Fragen, die die Studierenden lösen und beantworten. Dazu greifen sie über einen *Internet Explorer* auf aktive Seiten (ASP.NET [4]) zu, die auf dem Server ausgeführt werden und ihrerseits über SQL (Structured Query Language [5]) mit der Datenbank kommunizieren. Die Fragen selbst liegen in der Datenbank zum größten Teil in Form von einfachem HTML [6] vor; lediglich die mathematischen Formeln sind in *MathML* [7] (vgl.

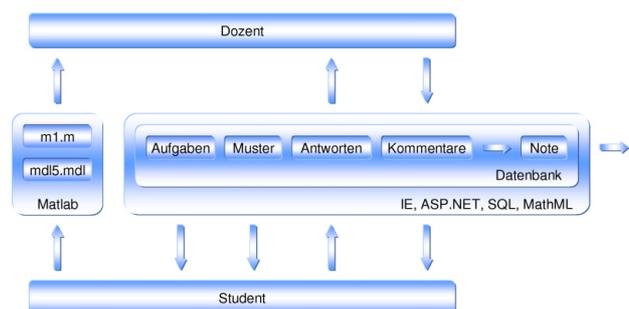


Abbildung 1: Kommunikation.

Abbildung 14) kodiert und werden von einem Browser-Plugin [8] interpretiert und dargestellt. Die Studierenden merken von dieser Hintergrundkommunikation natürlich nichts; für sie stellt sich das System als einfache Eingabemaske (vgl. Abbildung 4) dar.

Die Aufgaben selbst lösen die Studierenden mit *Matlab* (bzw. *SIMULINK*) [9]. Dabei nutzen sie sowohl dessen numerische Befehle (*roots*, *eig*, ...) als auch die analytischen Methoden (*solve*, *simple*, ...) der *Symbolic Math Toolbox* [10]. Alle Lösungen dokumentieren sie in entsprechenden Skript(.m)- bzw. Modell(.mdl)-Dateien. Parallel zum Erstellen der Lösungsdateien beantworten sie in der Lernumgebung Fragen zu den Lösungsstrategien. Diese Antworten werden direkt in der Datenbank gespeichert. Während des Erstellens der Lösungen und Antworten stehen den Studierenden unter genau definierten Randbedingungen, Musterlösungen und Musterantworten zur Verfügung.

In der Prüfungsphase am Ende des Semesters greift der Dozent auf die gesammelten Lösungsdateien und Antworten zu und schreibt seinerseits Kommentare und Punktabzüge in die Datenbank, die wiederum von den Studierenden eingesehen werden können. Aus den Punktabzügen wird automatisch eine Note für jeden einzelnen Studierenden erzeugt und vom fachbereichs-eigenen Notenerfassungsprogramm [11] veröffentlicht.

Gruppenarbeit

Ein ausgesprochen wichtiger Aspekt des Übungserfolgs ist das Arbeiten in Kleingruppen (zwei, in Ausnahmefällen drei Gruppenteilnehmer). Viele aktuelle Untersuchungen untermauern die These, dass das Lernen und selbst das Arbeiten am Rechner (*Extreme Programming* [12]) zu zweit deutlich effektiver geschieht als im Einzelkämpfermodus. Neben dem en passant-Erwerb der klassischen *soft skills* (Verantwortungsbewusstsein, Teamfähigkeit, ...) und der Tatsache, dass die Übungen zu zweit einfach mehr Spaß machen, trägt insbesondere die Notwendigkeit, über die Lösungsstrategien und die Antworten miteinander diskutieren zu müssen, zur Verfestigung des Gelernten bei. Dem Dozenten fällt in diesem Zusammenhang während der Übungen neben dem Beantworten fachlicher Fragen die wichtige Aufgabe zu, die Gruppenmitglieder immer wieder zu diesem wichtigen Dialog zu ermuntern und Aufteilungsbestrebungen (*Du denkst - ich schreibe.*) möglichst früh, freundlich aber bestimmt, zu unterbinden.

Prüfungsleistung

Da die Übungen der zentrale Bestandteil des Moduls sind, liegt es nahe, diese als Prüfungsleistung zu

bewerten. Eine Klausur *mit Papier und Bleistift* erscheint angesichts der rechnerorientierten Lehr- und Lernform völlig absurd und eine zeitgleiche Prüfung aller Studierenden am Rechner scheitert (momentan) an 15 *Matlab*-Lizenzen und 90 Studierenden. Außerdem würde eine punktuelle Prüfung am Ende des Semesters den Wert der wöchentlichen Lösungen und Antworten stark herabsetzen und die Studierenden während der Übungsbearbeitung demotivieren.

Um der Prüfungsordnung Genüge zu tun, in der gefordert wird, dass der Beitrag jedes Gruppenmitglieds klar erkennbar sein muss, kann auf Wunsch einzelner Gruppenmitglieder oder des Dozenten (wenn während der Übungen klar wurde, dass einzelne Gruppenmitglieder nicht das Niveau der Gruppe erreichen) eine mündliche Überprüfung jedes Gruppenmitglieds und die Differenzierung in Einzelnoten stattfinden.

Es gibt Kollegen, die befürchten, das Mathematik-Modul würde durch den Wegfall der Klausur am Ende des Semesters und die Umstellung auf Übungsbetrieb vom klassischen *Siebfach*, mit dem ungeeignete Studierende schon in den ersten Semestern herausgefiltert werden können, zum *dünnen Brett* mutieren, das mit Hilfe der Musterlösungen ja wohl jeder mit 1,0 bestehen kann.

Die Prüfungsergebnisse des ersten Durchgangs sprechen eine andere Sprache.

Wie in Abbildung 2 dargestellt ist, ergibt sich auch bei dieser Prüfungsform die übliche Glockenkurve mit einer 15%-igen Durchfallquote. Lediglich die hohe Anzahl von 4,0 fällt etwas aus dem Rahmen. Solche Häufungen diesseits der *gerade noch bestanden* - Grenze können beispielsweise dann entstehen, wenn ein wohlwollender Dozent bei der erneuten Durchsicht der *jenseitigen Fälle* doch noch den einen oder anderen Punkt findet...

STUDIERENDE

Login

Abbildung 3 zeigt die Login-Seite der Studierenden [13]. Diese wählen aus Auswahllisten das Modul, das

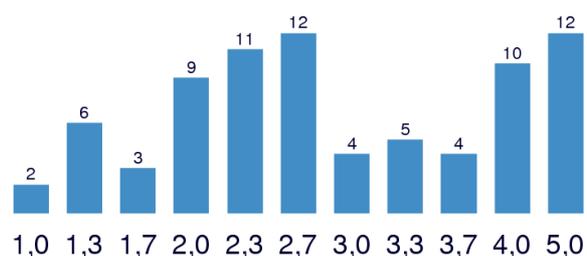


Abbildung 2: Notenverteilung der IMAT1-Prüfung vom 13.2.2004 [11].

Login

Modul
 Dozent
 Gruppe
 Kennwort

Hinweise

Zum Durchführen dieser Übungen verwenden Sie bitte einen Internet Explorer (Version 5.5 oder neuer) und installieren das MathPlayer-Plugin von <http://www.dessci.com/en/products/mathplayer/welcome.asp>, um die MathML-Formeln vernünftig darzustellen.

Außerdem müssen Sie Ihrem Browser das Annehmen von Cookies erlauben, was im IE standardmäßig eingestellt ist.

Abbildung 3: Login.

sie bearbeiten möchten, die entsprechende Dozentenabkürzung und ihre Gruppe aus. Da der Zugang zu den einzelnen Gruppen natürlich kennwortgeschützt ist und um die Dozenten von der Last zu befreien, alle Gruppen persönlich eintragen zu müssen, existiert immer eine Gruppe namens *dummy* mit dem Kennwort *dummy*, mit der sich eine potenzielle Gruppe erstmalig anmelden kann, um dann selbstständig eine neue eigene Gruppe zu eröffnen (*Münchhausen-Effekt*).

Des Weiteren informiert die Login-Seite darüber, dass alle mathematischen Formeln auf den Übungsseiten in *MathML* gesetzt sind und dass zu ihrer Darstellung ein kostenloses Browser-Plugin geladen werden muss.

Zentrale Übungsseite

Nach erfolgreichem Einloggen befinden sich die Studierenden auf der zentralen Übungsseite (Abbildung 4). In der Kopfleiste wird links das aktuelle Modul, der Dozent und der Gruppenname angezeigt. Den restlichen Platz der Kopfleiste nehmen die persönlichen Daten der Gruppenmitglieder ein. Jedes Mitglied ist dort mit seinem Namen, seiner E-Mail-Adresse, seiner Matrikelnummer und einem kleinen Porträt vertreten. Diese Kopfleiste ist während der Übungsbearbeitung immer sichtbar, was dem Dozenten das persönliche Ansprechen der Gruppenmitglieder und das *Lernen* ihrer Namen sehr erleichtert.

IMAT1
 BUCH
 Fiyerjuppels

Björn Krischker
 Juppel@gmx.de
 130782

Andreas Theissen
 andreas-theissen@freenet.de
 117262

Kapitel 14 Kegelschnitte
 Aufgabe 1 Es handelt sich um den Kegelschnitt "Kreis", da die gegebene Gleichung eine Kreisgleichung ist.

Schreiben Sie die Gleichung $(x-1.5)^2 + (y+1)^2 = 9$ in impliziter Form als $F(x,y) = 0$ und definieren Sie $F(x,y)$ unter MATLAB.
 Um welchen Typ von Kegelschnitt handelt es sich?
 Woran können Sie dies erkennen?
 Wo liegt er?
 Wie groß ist sein...?

Es handelt sich um den Kegelschnitt "Kreis", da die gegebene Gleichung eine Kreisgleichung ist, d.h. die Koeffizienten von "x" und "y" sind gleich, in unserem Fall "1".
 Er ist vom (0, 0) Punkt des Koordinatensystems um "1,5" nach rechts und um "1" nach unten verschoben. Sein Radius beträgt "3".

Abbildung 4: Zentrale Übungsseite.

Unter der Kopfleiste befindet sich der Navigationsbereich, in dem jeweils mittels zweier Schaltflächen und einer Auswahlliste das aktuelle Kapitel und die zu bearbeitende Aufgabe ausgewählt werden kann. Die Schaltflächen (*zurück* und *weiter*) erlauben dabei natürlich ein inkrementales *Blättern* zur vorherigen bzw. nächsten Aufgabe bzw. Kapitel, während die Auswahlliste alle Kapitel (etwa 20 Kapitel pro Modul) bzw. Aufgaben (etwa 10 Aufgaben pro Kapitel) *auf einen Blick* darstellt und somit ein direktes Anwählen des gewünschten Eintrags ermöglicht. Außerdem werden in der Aufgaben-Auswahlliste die (in den ersten Zeilen) schon eingetragenen Antworten dargestellt, so dass die Gruppe unmittelbar erkennen kann, ob möglicherweise noch eine Antwort im aktuellen Kapitel fehlt.

In der Mitte der Seite (unter dem Navigationsbereich) finden die Studierenden die aktuelle Aufgabe, Fragen zur Aufgabe und das Textfeld für ihre Antworten. Nach dem Eintragen der Antworten einer Aufgabe werden diese mittels der darunter angeordneten Schaltfläche *Antworten speichern* in der Datenbank gespeichert. Interessanterweise führte das in einer Vorgängerversion der Lernumgebung programmierte automatische Abspeichern der aktuellen Antwort beim Auswählen einer neuen Aufgabe (ohne explizite *Speichern*-Schaltfläche) zu Akzeptanz- und Sicherheitsproblemen bei den Studierenden; nicht zuletzt vielleicht auch deshalb, weil ein *Untätigkeits-timeout* zu einem automatischen Ausloggen und damit zu einem Verlust der noch nicht gespeicherten Antworten führen kann.

Auf der linken Seite des Fensters befindet sich ein Satz von Schaltflächen, deren Funktionen in den folgenden Kapiteln erläutert werden.

Anwesenheitskontrolle

Da das didaktische Konzept des Übungsseminars darauf beruht, dass die Studierenden die Übungsaufgaben in Anwesenheit der Dozenten lösen, muss

es eine effektive Form einer motivierenden Anwesenheitskontrolle geben. Die manuelle Auswertung von Unterschriftenlisten in Papierform stellt bei 30 Übungsterminen pro Semester eine ziemlich lästige Aufgabe dar; eine sichere digitale Anwesenheitsüberprüfung ist ohne größeren technischen Aufwand allerdings auch nicht wirklich einfach zu realisieren. Einem täuschungswilligen Studierenden ist es schließlich leicht möglich, einen beliebigen Kommilitonen mit seinen Kennwörtern oder anderen *persönlichen Merkmalen* auszustatten, damit dieser ihn als anwesend *einträgt*.

In MAUS wurde daher ein einfaches aber wirkungsvolles Anwesenheitsüberprüfungskonzept implementiert: Nach dem Einloggen wird statt der Bilder der Studierenden in der Kopfleiste nur ein markantes *Ich bin nicht anwesend*-Logo dargestellt (vgl. Abbildung 4). Durch Anklicken seines Logos dokumentiert dann jedes Gruppenmitglied seine Anwesenheit, worauf jeweils das persönliche Bild des einzelnen Mitglieds erscheint. Während der Übung ist es daher dem Dozenten jederzeit möglich, die Identität der Mengen der Gesichter diesseits und jenseits des Bildschirms zu verifizieren und bei Differenzen mahnende Worte zu sprechen.

Um wenigstens die offensichtlichsten Cheats auszuschließen (im eigenen Bett aufwachen, feststellen, dass jetzt eigentlich Übung ist, per Laptop mal kurz alle Gruppenmitglieder als anwesend eintragen, wieder hinlegen oder, nachdem der Dozent den Übungsraum verlassen hat, noch schnell die fehlenden Gruppenmitglieder nachmelden,...) kann die Anwesenheitsmeldung nur von einem Rechner in der Hochschule und nur während der ersten halben Stunde nach Übungsbeginn erfolgen. Die zweite Einschränkung motiviert gleichzeitig auch ein wenig dazu, *rechtzeitig* zur Übung zu erscheinen. Um späteren Diskussionen vorzubeugen, werden außerdem natürlich das Datum, die Uhrzeit und die IP-Adresse des benutzten Rechners mitprotokolliert.

Neue Gruppe Anlegen

Nachdem sich eine potenzielle Gruppe als *dummy* eingeloggt hat, kann sie ihre eigene Gruppe ins Leben rufen, indem sie auf der zentralen Übungsseite die Schaltfläche *Neue Gruppe* anklickt.

Auf der in Abbildung 5 dargestellten Seite kann dann eine neue Gruppe mit einem frei wählbaren Gruppenkennwort angelegt werden. Existiert der gewünschte Gruppenname bereits, wird der Gruppe vor einem erneuten Versuch eine Liste aller vorhandenen Gruppennamen angezeigt.

Neue Gruppe

Hinweis

Nachdem die neue Gruppe eingerichtet wurde, tragen Sie bitte die Gruppenmitglieder ein.

Abbildung 5: Neue Gruppe anlegen.

Gruppenmitglieder Definieren

Nach dem Erzeugen einer neuen Gruppe werden sinnvollerweise die Gruppenmitglieder eingetragen (Abbildung 6). Dabei gibt jedes Mitglied seinen Namen, seine E-Mail-Adresse und seine Matrikelnummer an und legt, wenn sein Datenschutzbedürfnis ihm dies erlaubt, sein beschriebenes Konterfei auf dem Server ab.

Gruppenmitglieder

Hinweise

- Jedes einzutragende Mitglied muss eine Matrikelnummer besitzen. Wenn Sie Ihre Matrikelnummer noch nicht kennen, tragen Sie bitte die Zahl 0 ein und ändern den Eintrag, sobald Sie Ihre Matrikelnummer kennen.
- Mitglieder können nur vom Dozenten wieder gelöscht werden.
- Das Bild muss eine JPG-Datei sein und sollte möglichst eine Höhe von 70 Pixeln haben.
- Jedes Mitglied kann nur einmal ein Bild hochladen.
- Ein Bild kann erst hochgeladen werden, wenn das Mitglied in die Datenbank eingetragen wurde.

Abbildung 6: Gruppenmitglieder definieren.

Gebräuchliche Matlab-Befehle

Obwohl jeder neue *Matlab*-Befehl in der entsprechenden Übung eingeführt wird und ab dann eigentlich als bekannt vorausgesetzt werden kann, ist *Wie heißt noch mal der Befehl, um...* wohl die am häufigsten gestellte Frage während der Übungen. Auch syntaktische *Feinheiten* *Muss ich da jetzt ein Komma oder...* oder *Ich dachte, eckige Klammern...* sind gerade in der Anfangszeit für viele Studierende ernste Gegner.

Aus diesem Grund führt das Anklicken der Schaltfläche *Matlab* auf der Übungsseite zu der in Abbildung 7 dargestellten Seite und trägt damit einem vielfach geäußerten Wunsch der Studierenden Rechnung.

Die meisten der in den Übungen verwendeten Befehle sind hier im Rahmen eines charakteristischen Beispiels aufgelistet, so dass ein Suchvorgang (Strg-F) auf der Seite schnell zum *vergessenen* Befehl führt; vorausgesetzt, man hat zumindest eine ungefähre Vorstellung davon, wonach man eigentlich sucht...

Matlab-Beispiele

Beispiel	Erläuterung
[]	Leere Matrix
[3, 4.2, pi] oder [3 4.2 pi]	Zeilenvektor (Elemente durch Komma oder Leerzeichen getrennt)
[3, 5; 8, 2; 4, 0]	Matrix (3 Zeilen und 2 Spalten)
[78; 42; 7; 555]	Spaltenvektor (Elemente durch Semikolon getrennt)
[echt, rest] = proper (bruch)	Aufspalten von bruch in seinen echt gebrochenen Anteil und den Rest
[V, D] = eig ([1 2; 3 4])	Ermitteln der Eigenvektoren (V) und der Diagonalmatrix der Eigenwerte (D) der Matrix [1 2; 3 4]
[winkel, radius] = cart2pol (x, y)	Kartesische Koordinaten (x,y) in Polarkoordinaten (winkel und radius) umwandeln
[z, n] = numden (bruch)	bruch in zaehler (z) und nenner (n) aufspalten
{'rot', 'gruen', 'blau'}	Menge mit den Elementen rot, gruen und blau
0 : 0.2 : 5	Vektor mit den Elementen [0, 0.2, 0.4, ..., 4.8, 5.0]
3^4	Potenz ("3 hoch 4")
a < b	1 (wahre Aussage), wenn a kleiner als b ist und 0 (falsche Aussage), wenn a größer als b ist
a == b	1 (wahre Aussage), wenn a identisch mit b ist und 0 (falsche Aussage), wenn a nicht identisch mit b ist
a(:,2)	Zweite Spalte der Matrix a
a(3,:)	Dritte Zeile der Matrix a
A\b	Lösen des linearen Gleichungssystems mit der Koeffizientenmatrix A und dem Vektor der rechten Seite c
abs (-42)	Betrag von -42
atan (1)	Arkustangens von 1
axis ([0 3 -2 2])	Nachträgliches Einschränken des dargestellten Bereichs auf 0 < x < 3 und -2 < y < 2
axis equal	gleiche Achsenskalierung auf x- und y-Achse
bar3 (matrix)	Balkendiagramm einer Matrix
cameratoolbar	Einschalten der cameratoolbar im aktuellen Bild, z.B. um die Blickrichtung zu verändern

Abbildung 7: Gebräuchliche *Matlab*-Befehle.

E-mail Versenden

Zur Erhöhung der Datensicherheit kann sich jede Gruppe durch Betätigen der Schaltfläche *E-Mail* auf der Übungsseite alle ihre Antworten an die eingetragenen E-Mail-Adressen aller Gruppenmitglieder senden lassen (Abbildung 8).

Die gesendete E-Mail enthält die Antworten als Text im Ascii-Format:

```
IMAT1 - Antworten der Gruppe
„Flyerjuppels“
```

```
1.1 (07.01.2004 , 16:41:29)
Matlab rechnet nach der Regel „Punkt
vor Strich-Rechnung“.
```

Wir setzten(3+4) in Klammern setzen.

```
1.2 ***** Noch nicht
beantwortet! *****
```

```
1.3 (07.01.2004 , 16:51:55)
„doc“ ist eine ausführliche
Dokumentation. es wirkt durch die
optische Aufmachung übersichtlicher
(z.B. Rechenbeispiele).
Nachteil: Es wird ein neues Fenster
geöffnet. „help“ ist die Kurzform
```

```
und etwas unübersichtlich.
1.4 (07.01.2004 , 17:00:46)
Ja.
Matlab verwendet das Bogenmass.
sin(90°) entspricht also sin(pi/2)!
```

Aufgaben, die noch nicht bearbeitet wurden, werden speziell gekennzeichnet, so dass die Studierenden sie *auf einen Blick* erkennen können.

E-Mail

Ihre Antworten werden an folgende Mitglieder der Gruppe gesendet:

1. Björn Krischker (Juppel@gmx.de)
2. Andreas Theißen (andreas-theissen@freenet.de)

Abbildung 8: E-Mail versenden.

MUSTERLÖSUNG UND MUSTERANTWORT

Die Tatsache, dass direkt von der Übungsseite (Schaltfläche *Muster*) zu jeder Aufgabe eine Musterlösung und zu jeder Frage eine Musterantwort aufgerufen werden kann, löst bei vielen Studierenden ungläubiges Staunen und bei Kollegen teilweise vehemente Ablehnungsäußerungen hervor. Wer

allerdings schon mal in studentische Mailinglisten hineingeschnuppert hat, findet dort von nahezu jedem Laborbericht mehrere als gut befundene *Alte Meister*, aus denen die aktuelle Version mit Copy-and-Paste und ein paar Datenanpassungen ruck-zuck zusammengeklickt ist. Die Hoffnung, Berichte oder Referate über etablierte Themen würden noch von den Studierenden selbst ausgearbeitet, überschreitet die Grenze der Naivität deutlich.

Die einzige Möglichkeit, diesen Plagiatismus wirkungsvoll zu unterbinden, wäre ein alljährlich komplett neu erstellter Satz von Übungen, Aufgaben und Fragen. Wenn man allerdings berücksichtigt, dass der Faktor zum Erstellen einer rechnerbasierten Lerneinheit zwischen 10 und 20 liegt, für eine einzige Semesterwochenstunde also bis über 200 Zeitstunden Vorbereitung nötig sind, verliert auch dieser Ansatz deutlich an Attraktivität.

Vor diesem Hintergrund dieses Dilemmas erscheint es konsequent und ehrlich, gleich mustergültige Lösungen und Antworten zu veröffentlichen (Abbildung 9), die den Vorteil haben, immer aktuell zu sein, also auch bei kurzfristigen Änderungen der Aufgabenstellung unmittelbar nachgeführt zu werden.

Da ein Großteil der Übungen in Anwesenheit der Dozenten durchgeführt wird und die Studierenden gehalten sind, erst ihre eigenen Lösungen und Antworten zu formulieren und erst dann optional die Musterlösungen bzw. -antworten zu konsultieren führt dieses Konzept zu einem erstaunlich eigenständigen Arbeitsverhalten der einzelnen Gruppen.

REGELN

Die besondere Veranstaltungs- und Prüfungsform gebietet es, die Studierenden sowohl in mündlicher als auch in schriftlicher Form über die besonderen Randbedingungen der Übungsdurchführung zu informieren. Zu diesem Zweck existiert auf der Übungsseite die *Regeln*-Schaltfläche, die zu der in Abbildung 10 dargestellten Seite führt.

Neben formalen Richtlinien zur Nomenklatur und Abgabeform findet sich hier eine ausführliche Beschreibung der gewünschten Verwendung von Musterlösung und -antwort und ein Sicherheitshinweis.

Musterlösung und Musterantwort

Musterlösung

[m14.m](#)

Musterantwort

Da die Koeffizienten der quadratischen Glieder 1 betragen, handelt es sich um einen Kreis. Sein Mittelpunkt ist um 1,5 nach rechts und um 1 nach unten verschoben und sein Radius beträgt 3.

Ja - natürlich weiß ich, dass ich aus der Musterlösung und der Musterantwort nichts kopieren darf...

[Danke](#)

Abbildung 9: Musterlösung und Musterantwort.

Regeln zur Übungsbearbeitung

- Legen Sie Ihre eigenen Lösungen bitte unter den Dateinamen `m1.m, ..., m8.m, ..., md19.m` in einem Verzeichnis Ihrer Wahl ab. Dabei benennen Sie bitte alle Lösungen genau nach dem vorgegebenen Schema:
 - Matlab-script-Dateinamen beginnen mit `m`, dann folgt die Kapitelnummer und dann die Dateierendung `.m`.
 - Simulink-model-Dateinamen beginnen mit `md`, dann folgt die Kapitelnummer und dann die Dateierendung `.mdl`.
- Verwenden Sie bitte genau die angegebene Nomenklatur. Wirklich – genau diese! Mein automatischer Ablaufalgorithmus hat nämlich ziemlich Schwierigkeiten, sich Ihre Lösungen aus Dateien wie `antwort_1.m, m11_test.m, mat1ab1.mat` oder `loesung3` zusammen zu suchen.
- Zippen Sie bitte am Ende des Semesters alle Ihre Lösungsdateien zusammen und senden mir die zip-Datei per E-Mail. Nennen Sie in der E-Mail bitte Ihren Gruppennamen.
- Ein paar Worte zum leidigen *ich darf nicht abschreiben*-Thema: Ich gehe davon aus, dass Sie im Rahmen dieser Übungen etwas über die Grenzen und Schwierigkeiten der Umsetzung theoretischer Konzepte in die Praxis lernen möchten. Andererseits haben Sie aber selbst schon oft genug die Erfahrung gemacht, dass Sie Methoden, Algorithmen und Verfahren am besten dann lernen, wenn Sie die unvermeidlichen Fehler beim Lösen der Aufgaben selbst machen. Seien Sie also fair zu sich selbst und gönnen Sie sich den Genuss eines eigenen Erfolgserlebnisses. Das Vorgehen mit dem größten Lernerfolg sieht daher in der Regel etwa so aus:
 - Übungsaufgabe lesen und verstehen.
 - Übungsaufgabe lösen und Fragen beantworten.
 - Dann erst Musterlösungen ausführen und Musterantworten lesen.
 - Bei großen Diskrepanzen zwischen Ihrer Lösung und der Musterlösung entweder Ihre Fehler korrigieren oder begründen, warum die Musterlösung falsch ist.
- Das Kopieren (auch einzelner Zeilen oder Blöcke) aus der Musterlösung muss daher konsequenterweise als *Täuschungsversuch* gewertet werden.
- Durch Anklicken des Bildes links neben Ihrem Namen bestätigen Sie, dass Sie in der jeweils aktuellen Übung anwesend sind. Führen Sie diese Anwesenheitsbestätigung bitte nur während der Übung im Übungsraum von einem der HSB-Rechner (kein Laptop) durch.
- Bearbeiten Sie bitte niemals die Antworten einer Gruppe gleichzeitig auf mehreren Rechnern und bearbeiten bitte niemals die Antworten mehrerer Gruppen gleichzeitig auf einem Rechner. Benutzen Sie bitte niemals die Zurück-Schaltfläche Ihres Browsers. Bei Nichtbeachtung können Ihre Antworten verloren gehen.

[Erverstanden](#)

Abbildung 10: Regeln.

KOMMENTARE UND NOTEN

Wie oben in *Korrigieren* beschrieben, korrigiert der Dozent die Lösungen und Antworten der Studierenden, indem er Kommentare einfügt und Punktabzüge (Notenverschlechterungen) vergibt, aus denen automatisch die Endnote der Gruppe berechnet wird. Nach Ende der Korrektur stehen diese Informationen für jede Gruppe auf einer eigenen Seite zur Verfügung, die mittels der Schaltfläche *Ergebnis* erreicht werden kann. Wie man in Abbildung 11 sieht, kann der Dozent auch Kommentare ohne Notenverschlechterung abgeben.

Aufgabe 29.4
 Entwickeln Sie $f(x)$ und $g(x)$ in eine Maclaurinsche Reihe.
 Beschreiben Sie die Reihen. Welche Potenzen sind vorhanden? Erklären Sie ein Bildungsgesetz für die Koeffizienten! Warum ist es nicht möglich, ein Lösungsprogramm zu schreiben, das für jede beliebige Funktion analytisch den Konvergenzradius der zugehörigen Reihenentwicklung erhebt?
 In jedem Summanden der Reihen ist ein Bruch vorhanden. Im Zähler steht jeweils eine Potenz von "x" und in der Nenner eine Fakultät. Bei der Exponentialfunktion liegen alle natürlichen Zahlen als Potenzen vor. Bei der Cosinus Funktion sind es die geraden natürlichen Zahlen und bei der Sinus Funktion sind es die Ungeraden, die als Potenzen vorliegen.
 richtig?
 ----- Notenverschlechterung um 3

Aufgabe 29.13
 Führen Sie die notwendige Umformung von y durch, um die de L'Hôspitalsche Regel anwenden zu können. Warum funktioniert `simple` nicht?
 Der Befehl "simple" führt zwar die nötige Rechenoperation aus, um den Nennernenner zu bilden, allerdings wird dies von Matlab nicht als Vereinfachung "erachtet". Deshalb führt "simple" nicht zum Ergebnis, sondern nur der explizite Befehl "factor".
 simple sucht den Ausdruck mit den wenigsten Zeichen.
 ----- Notenverschlechterung um 3

Aufgabe 29.17
 Berechnen Sie $\int_0^1 (1-x)^2 dx$.
 Welches interessante Ergebnis erhalten Sie? Von welchem Typ ist der erhaltene Ausdruck? Philosophieren Sie ein wenig über die Frage, was i^i denn nun wirklich ist (3, 1, 42, π , NaN, ...).
 Wir erhalten das Ergebnis "e". "i" unendlich". Mit Matlab erhalten wir das Ergebnis "NaN". "i" ist "i" sehr viele Male mal eine ergibt immer noch eins. Insofern wästen unendlich viele Male immer noch eins ergeben. Wahrscheinlich liegt es an mystischen Ausdruck "unendlich", der einen auch zur Wiedergeburt dieser These zwingen kann.....aber
 Wie sieht es mit 0^0 aus?
 ----- Notenverschlechterung um 3

Gesamtnote: 1,3
[Zurück](#)

Abbildung 11: Kommentare und Noten.

Logout

Um sicherzustellen, dass nachfolgende Rechnerbenutzer nicht auf ihre Übungsdaten zugreifen können, sollte jede Gruppe am Ende der Übung die *Logout*-Schaltfläche der Übungsseite benutzen.

Nach 20 Minuten *Untätigkeit* loggt das System die Gruppe automatisch aus. Wenn nun also eine Gruppe länger als 20 Minuten mit *Matlab* arbeitet, diskutiert, Antworten formuliert und in dieser Zeit keine Schaltfläche auf der Übungsseite betätigt, ...

DOZENTEN

Dozenten-Login

Auf seiner Login-Seite (Abbildung 12) wählt der Dozent lediglich das zu korrigierende Modul und sein Kürzel aus und legitimiert sich durch sein Kennwort. Korrigieren

Der Aufbau der Korrektur-Seite (Abbildung 13) ist dem der Studierenden-Übungsseite sehr ähnlich. In der Kopfleiste werden links das Modul und der Dozent und zusätzlich die aus der Summe der Punktabzüge berechnete momentane Gesamtnote der gerade begutachteten Gruppe dargestellt. Rechts davon befinden sich die Daten der Gruppenmitglieder.

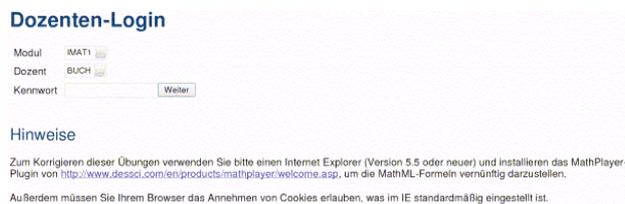


Abbildung 12: Dozenten-Login.



Abbildung 13: Korrigieren.

Im darunter liegenden Navigationsbereich kann jetzt nicht nur das zu beurteilende Kapitel und die Aufgabe sondern auch die Gruppe ausgewählt werden. Damit ist auch hier das beim Korrigieren von Klausuren übliche Vorgehen möglich, nicht alle Aufgaben einer einzelnen Klausur (Gruppe) nacheinander sondern zeitnah jeweils die gleiche Aufgabe in allen Klausuren (Gruppen) zu korrigieren, um eine größere Beurteilungsgerechtigkeit zu gewährleisten.

Unter dem Navigationsbereich wird die Antwort der aktuellen Gruppe zur aktuellen Frage in einem schreibgeschützten Textfeld genau so dargestellt, wie die Gruppe sie eingegeben hat.

Darunter kann der Dozent seinen Kommentar zur aktuellen Aufgabe in ein weiteres Textfeld eintragen und in der nächsten Zeile optional einen Punktabzug (Notenverschlechterung) wählen, der dann automatisch zur Gesamtnote (oben links) addiert wird.

Ganz unten auf der Seite wird zur Erinnerung der Text der aktuellen Aufgabe, Frage und Musterantwort dargestellt. Üblicherweise braucht der Dozent beim Korrigieren einer Aufgabe diesen Text natürlich nur bei den *ersten paar* Gruppen.

FRAGEN UND ANTWORTEN MODIFIZIEREN

Alle Fragen und Antworten eines Moduls eines Dozenten befinden sich in einer Datenbank (z.B. imat1_buch.mdb) auf dem Server. Wenn der Dozent die Werkzeuge zum Modifizieren der Daten in der Datenbank beherrscht, ist es bei größeren Änderungen natürlich am einfachsten, direkt in der Datenbank zu arbeiten. Wenn nicht – bietet die Schaltfläche *Inhalt* auf der Korrekturseite die Möglichkeit, wenigstens kleine Textmodifikationen über eine Web-Schnittstelle durchzuführen.

Die in Abbildung 14 dargestellte Seite besteht neben der üblichen Navigationsstruktur aus drei Textfeldern, in denen die aktuelle Aufgabe, Frage und Antwort direkt modifiziert werden kann. Deutlich erkennt man im Fragenfeld die HTML-Syntax (`
`) und im Aufgabenfeld die in *Studierende* angesprochene *MathML*-Kodierung (`<m:math>`) der mathematischen Formeln, die später vom MathPlayer-Plugin des Internet Explorers dargestellt werden. Der *MathML*-Quelltext selbst wird dabei nur von echten Hardcore-Designern noch manuell kodiert; Durchschnittsdozenten nutzen lieber übersichtliche grafische Werkzeuge (wie z.B. *MathType* [14]), um beliebige Formeln *wie in* Word komfortabel zusammenzuklicken und als *MathML* über die Zwischenablage direkt in das entsprechende Textfeld zu exportieren.



Abbildung 14: Fragen und Antworten modifizieren.

ÜBERSICHT

Auf der Übersichtsseite (Abbildung 15) findet der Dozent in drei Tabellen die persönlichen Daten, die Noten und die Anwesenheitszeiten der einzelnen Studierenden. Die Tabellen lassen sich nach jeder Spalte sortieren, so dass beispielsweise die Studierenden nach Matrikelnummer, Vorname, Nachname, E-Mailadresse oder nach ihrer Gruppenzugehörigkeit sortiert werden können.

In der zweiten Tabelle sind nur die Matrikelnummern der Studierenden und ihre Noten aufgelistet. Obwohl die Noten natürlich auch einfach in einer weiteren Spalte der Studierenden-Tabelle geführt

Studierende				
Matrikelnummer	Vorname	Nachname	Email	Gruppe
131300	Tilmann	Basko	muthey@gmx.de	ALMA
127593	Manuel	Beck	mbeck@fbm.hs-bremen.de	fly2007
136055	Ulrich	Beinert	analemma@gmx.de	Beikle

Noten	
Matrikelnummer	Note
10	2
29856	3,3
77978	2,7
78350	1

Anwesenheit				
Vorname	Nachname	Termin	Rechner-IP	Eingelgt
Alexander	Geist	25.02.2004 15:00:00		25.02.2004 14:10:20
Alexander	Haupt	25.02.2004 15:00:00		25.02.2004 14:09:28
Alexander	Haupt	26.02.2004 09:30:00	192.168.116.33	26.02.2004 09:34:22
Andreas	Theißen	26.02.2004 09:30:00	192.168.116.33	26.02.2004 09:34:41

Abbildung 15: Übersicht (Ausschnitte).

werden könnten, hat die Auslagerung in eine separate Tabelle den Vorteil, dass genau diese Information (Matrikelnummer und Note) für die Notenveröffentlichung benötigt wird. Beispielsweise lässt sich der Inhalt der Noten-Tabelle einfach per Drag-and-Drop in das Notenerfassungsprogramm des Fachbereichs Maschinenbau der Hochschule Bremen übernehmen, das alle Prüfungsergebnisse statistisch und grafisch aufbereitet an zentraler Stelle im Netz verfügbar macht [11].

Die dritte Tabelle protokolliert die Anwesenheit der Studierenden an den einzelnen Übungsterminen. Momentan wird dabei pro studentischer Anwesenheitsmeldung eine Zeile mit Name, Zeitpunkt und Rechner-IP ausgegeben. Eine grafische Aufbereitung der Daten ist in Arbeit.

LESSON LEARNED

Um Rückmeldungen über ihre Lehrqualität zu erhalten, bitten einige Dozenten die Studierenden darum, ihr Modul im Rahmen einer netzbasierten Evaluation (Online-Fragebogen) zu beurteilen [15]. Die aussagekräftigsten Informationen stellen dabei sicherlich die ausformulierten Antworten auf Fragen wie *An der Veranstaltung (bzw. am Dozenten) finde ich schlecht* dar. Die folgenden Originalkritiken (Rechtschreibfehler korrigiert) spiegeln recht gut die Probleme und Stärken des ersten Durchgangs der rechnerbasierten Mathematikübungen wider:

- *Es war zu sehr auf Matlab basierend. Er hat uns wenig über das Programm aufgeklärt. In den praktischen Stunden am Computer wurde wenig das Inhaltliche gefordert, stattdessen musste man sich mehr mit der Ausführung des Programms auseinander setzen. Das meiste vom Inhalt war bereits aus der Schule bekannt.*
- *Keine Einführung in Matlab.*
- *Durch die Konzentration auf Matlab kommt das eigentliche Verständnis für die mathematischen Sachverhalte bei vielen bestimmt zu kurz.*
- *Leider fällt die Mathematik bei vielen Studierenden im komplexen Umgang mit Matlab vom Tisch.*
- *Diese Veranstaltung gleicht meiner Meinung nach sehr einer Informatik-Veranstaltung, da man sehr viel Zeit benötigt, um sich mit Matlab auseinander zu setzen.*

Der Sprung ins kalte *Matlab*-Wasser stellt für einige Erstsemester einen heftigen mathematischen Kulturschock dar. Schon während der Übungen wurde deutlich, dass besonders die Studierenden, die

beispielsweise in der Schule einen (klassischen) Mathe-Leistungskurs mit teilweise sehr guten Noten absolviert aber noch wenig Erfahrung im Umgang mit Rechnern hatten, plötzlich feststellen mussten, dass Fertigkeiten wie *Partialbruchzerlegung* und *Gaußscher Algorithmus* oder gar Beweisverfahren wie *Vollständige Induktion* jetzt nicht mehr ausreichen, um erfolgreich Ingenieurmathematik zu betreiben. Teilweise führte diese Erkenntnis sogar zu temporären Lernblockaden, Ablehnen des Werkzeugs und gefühlsbetonten Unmutsäußerungen à la *Ich weiß doch, wie ich das Integral lösen kann; warum soll ich das auch noch der blöden Kiste beibringen?*

Sehr schön auf den Punkt bringt dieses Gefühl die folgende Kritik:

Durch das Ausrechnen von komplexen Aufgaben lernt man Konzentration und Disziplin. Man kann es sich einfach nicht erlauben, in einer dreiseitigen Rechnung unaufmerksam zu sein. Und genau das fehlt bei Matlab.

Nicht erst seit dem Hollywoodklassiker *Matrix* (der Aufstand der schlesischen Weber gegen die mechanischen Webstühle fand immerhin schon im Jahre 1844 statt) fürchten wir Menschen, dass Maschinen uns zwar satt und zufrieden aber auch arbeitslos und dumm machen. Der menschliche Geist ist daher ständig auf der Suche nach neuen Herausforderungen und der Bestätigung seiner eigenen Überlegenheit.

Wenn es dann allerdings um die Tragflächen eines A380 geht, benutzen wir Ingenieure für die Konstruktion doch lieber ein CAD-Programm als das gute alte Zeichenbrett und für die Produktion doch eher eine CNC-Fräse als die Feile. Und für die Profiloptimierung und die Aerodynamik-Simulation ziehen wir eben Programme wie *Matlab* der Handrechnung mit Papier und Bleistift vor, auch wenn wir dann nicht mehr jeden Rechenschritt im Einzelnen nachvollziehen können. Und bei Stromausfall gehen wir nach Hause...

Manche Studierenden erkennen den Wert einer rechnerbasierten Ingenieurmathematieausbildung unter der Überschrift *An der Veranstaltung (bzw. am Dozenten) finde ich gut schon recht früh:*

- *Sehr praxisorientiert, ..., Matlab Kenntnisse wahrscheinlich für den Beruf sehr hilfreich.*
- *Was ich auch gut finde, ist das PC-unterstützte Arbeiten.*
- *Ich finde die Laborstunden mit Matlab sehr interessant und es hat auch sehr viel Spaß gemacht.*

- *Gute Kombination von Mathematik mit der Anwendung am PC.*
- *Computerbasiertes Lernen.*

Wichtig ist es in diesem Zusammenhang also anscheinend, die Berührungängste der Studierenden mit dem unbekanntem Werkzeug ernst zu nehmen und nach Wegen zu suchen, den Einstieg in die digitale Mathematik so reibungsarm wie möglich zu gestalten. So entstand beispielsweise die Liste der *Matlab*-Befehle nach dem ersten Semester aus der Anregung und Vorarbeit eines Studierenden.

Oder aber man wartet, bis die Zeit alle Wunden heilt:

Zu Beginn (so die ersten 6 Wochen) hat mich Matlab schon etwas Nerven gekostet, aber wenn man ein wenig verstanden hat, wie Matlab denkt, läuft es besser.

REFERENZEN

1. Fachbereich Maschinenbau, *Homepage* (2004), <http://www.hs-bremen.de/Deutsch/Seiten.asp?SeitenID=44>
2. Internationaler Studiengang Luftfahrt-systemtechnik und -management, *Homepage* (2004), <http://www.hs-bremen.de/Deutsch/Seiten.asp?SeitenID=59>
3. Papula, L., *Mathematik für Ingenieure und Naturwissenschaftler*. Band 1 - Band 3, Vieweg (2001).
4. W3Schools, *ASP.NET Tutorial* (2004), <http://www.w3schools.com/aspnet>
5. W3Schools, *SQL Tutorial* (2004), <http://www.w3schools.com/sql>
6. W3Schools, *HTML Tutoria*, (2004), <http://www.w3schools.com/html>
7. The World Wide Web Consortium, W3C Math Home, (2004), <http://www.w3.org/Math/>
8. Design Science, Inc., *MathPlayer Download* (2004), <http://www.dessci.com/en/products/mathplayer/download.htm>
9. The Mathworks, *Matlab* (2004), <http://www.mathworks.com>
10. The Mathworks, *Symbolic Math Toolbox* (2004), <http://www.mathworks.com/products/symbolic/>
11. Fachbereich Maschinenbau, *Prüfungsergebnisse* (2004), http://www.fbm.hs-bremen.de/noten/auswahl_darstellung.asp
12. Wired, *The New X-Men* (2003), http://www.wired.com/wired/archive/11.09/xmen_pr.html
13. Buchholz, J.J., *Mathematik Übungsseminar*

(2004), <http://www.fbm.hs-bremen.de/maus/default.aspx>

14. Design Science, Inc., *MathType* (2004), <http://www.dessci.com/de/>
15. Fachbereich Maschinenbau, *Evaluation* (2004), http://www.fbm.hs-bremen.de/eva/auswertung_auswahl_fachbereich.asp

BIOGRAPHIE



Jörg J. Buchholz ist Professor für Ingenieurmathematik, Regelungstechnik und Flugregelung im Fachbereich Maschinenbau der Hochschule Bremen.

Er wurde am 14.11.1959 in Neumünster geboren. Im Jahre 1985 schloss er sein

Studium der Elektrotechnik an der Technischen Universität Braunschweig ab. Im Jahre 1991 promovierte er am Institut für Flugführung der Technischen Universität Braunschweig mit der Arbeit *Sensorfehlererkennung in Flugzeugen mit Beobachter und Polynomklassifikator*.

Von 1990 bis 1995 war er als wissenschaftlicher Mitarbeiter im Institut für Flugmechanik (heute Institut für Flugsystemtechnik) des Deutschen Zentrums für Luft- und Raumfahrt (DLR) auf den Gebieten Flugregelung, Flugmechanik, Modellierung, In-flight-Simulation, Modellfolgeregelung, Pilot Induced Oscillations, Genetische Algorithmen und Rekonfiguration aktiv.

Seit 1995 lehrt er als Professor an der Hochschule Bremen und beschäftigt sich mit Flugregelung, Modellierung, Simulation, Lenkdrachen, Mikrokontrollern, Mustererkennung, Parameteridentifikation und Kryptografie.