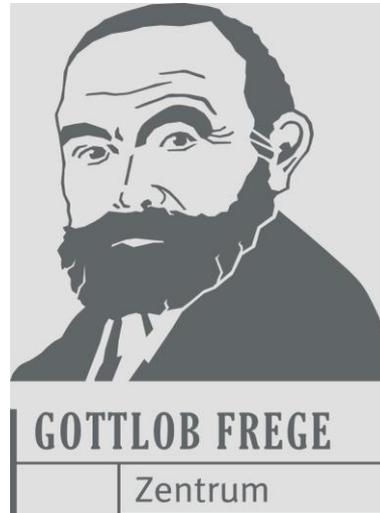


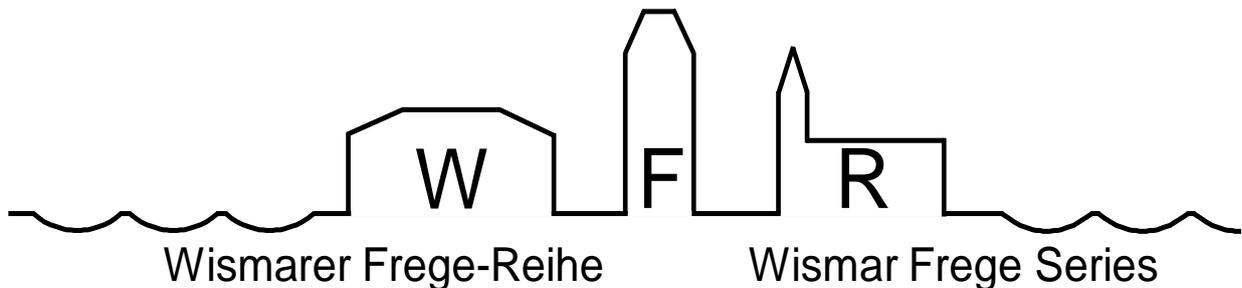
Hochschule Wismar Gottlob Frege Centre



Proceedings
18. Workshop Mathematik
in ingenieurwissenschaftlichen Studiengängen

Bochum
November 2023

Heft 02 / 2023



Das **Gottlob-Frege-Zentrum** wurde am 7.11. 2000 an der Hochschule Wismar gegründet. Seine Mitglieder setzen sich für eine wissenschaftlich begründete, praxisorientierte, moderne und international ausgerichtete Ausbildung in den mathematisch-naturwissenschaftlichen Grundlagendisziplinen ein.

Weitere Informationen zum Gottlob-Frege-Zentrum finden Sie im Netz unter

www.hs-wismar.de/frege

bzw. auf der Netzseite

<https://www.hs-wismar.de/vernetzung/institutionen-hochschulunternehmen/gottlob-frege-zentrum/>

Die **Wismarer Frege-Reihe** (WFR) veröffentlicht neben Beiträgen zur Frege-Tradition auch Beiträge zur Mathematikausbildung.

Sie ist urheberrechtlich geschützt. Eine Vervielfältigung ganz oder in Teilen, ihre Speicherung sowie jede Form der Weiterverbreitung bedürfen der vorherigen Genehmigung durch den Herausgeber.

ISSN 1862-1767

ISBN 978-3-947929-25-2

Alle Rechte vorbehalten.

© Hochschule Wismar 2023.

Printed in Germany

WFR Heft 02/2023

Inhalt / Contents

Vorwort	3
Anmerkungen des Herausgebers	4
Vortragsprogramm	5

Artikel / Articles

Historische Mathematiklehre

Dieter Schott: <i>Mathematiklehre an Schulen und Universitäten zu Zeiten von Gottlob Frege</i>	6
--	---

Computereinsatz in der Mathematiklehre

Jürgen Vorloeper: <i>Symbolisches und Numerisches Rechnen mit LuaLATEX</i>	13
--	----

Arbeiten mit Lern-Plattformen

Alexander Dominicus: <i>Automatisiertes Assessment iterativer Verfahren im Mathematikbereich in Moodle</i>	19
--	----

Torsten-Karl Stempel: <i>Automatische Aufgaben in Moodle</i>	25
--	----

Georg Hoever: <i>Integration von Mumie-Aufgaben in der höheren Mathematik-Ausbildung</i>	33
--	----

Lern-Applikationen

Patricia Graf, Heiko Knospe, Andreas Schwenk: <i>Mathe:buddy - eine Gamifizierte Lern-App für die Höhere Mathematik</i>	38
---	----

Malte Neugebauer: <i>Lightweight Learning Analytics Dashboard for Analyzing the Impact of Feedback & Design on Learning in Mathematical E-Learning</i>	44
--	----

Hildo Bijl: *Application of Performance Distribution Tracing to a Thermodynamics Course* 50

Hinweise: Die Beiträge in den Proceedings können einen anderen Titel als die Vorträge haben. Auch die Reihenfolge ist hier eine andere. Sie orientiert sich an übergreifenden Themen.

Die Vortragenden Herr Kallweit, Frau Schmitz und Frau Kaiser haben keine Beiträge eingereicht.

Anhang / Appendix

WFR - Übersicht 57

Vorwort

Die Workshop-Reihe „Mathematik in ingenieurwissenschaftlichen Studiengängen“ (IngMath) wird seit 2001 jährlich an verschiedenen Hochschulen ausgerichtet und dient dem deutschlandweiten Austausch zwischen Lehrenden der Mathematik, insbesondere im Ingenieurbereich. Die veranstaltenden Hochschulen legen die Themen für den Austausch im Voraus fest und geben sie rechtzeitig bekannt, sodass Teilnehmende ihre Vorschläge für Beiträge einreichen können. Der 11. Workshop fand im Jahre 2013 an der Hochschule Bochum statt. Wir freuen uns, dass die Hochschule Bochum nach einer Zeitspanne von 10 Jahren erneut Ausrichter des Workshops sein konnte.

Am 10. November 2023 trafen sich 34 Lehrende der Mathematik von verschiedenen Hochschulen aus Deutschland zum 18. Workshop. Die Themen des Workshops lauteten „Softwareintegration in den Mathematikunterricht (zum Üben und Programmieren)“ und „Alternative Unterrichtsformen und Bonussysteme“.

Prof. Dr. Dieter Schott führte kurzweilig mit historischen Anmerkungen zur „Mathematiklehre an Schulen und Universitäten zu Zeiten von Gottlob Frege“ zum Thema. Der Sprecher des Arbeitskreises „Digitale Mathematik-Aufgaben in der Hochschullehre“, Dr. Michael Kallweit von der Ruhr-Universität Bochum, hielt anschließend die Keynote „Einsatz digitaler Mathematikaufgaben: Innovative Lehransätze mit STACK“. Aus den eingereichten Vorschlägen konnten neun weitere Vorträge gehalten werden. In diesem Jahr stieß das Thema des Workshops auch über den Ingenieurbereich hinaus auf Interesse. So waren auch Mathematiklehrende aus wirtschaftswissenschaftlichen Studiengängen dabei. Nicht nur während der Vorträge fand ein reger Austausch statt, sondern auch in den Pausen waren die Gespräche und Diskussionen zwischen den Teilnehmenden so brennend, dass die Einhaltung der Agenda fast nicht möglich war. Der Workshop war ein voller Erfolg.

Mit der Veröffentlichung in der Wismarer Frege-Reihe sind die Inhalte der Beiträge nachlesbar und stehen einem größeren Publikum zur Verfügung. Das freut uns sehr.

Im Sommer 2024 wird der 19. IngMath-Workshop voraussichtlich an der Hochschule Bielefeld stattfinden.

Anna Prokop und Claudia Frohn-Schauß



Anmerkungen des Herausgebers

Im Namen aller Beteiligten danke ich Frau Prof. Dr. Frohn-Schauß und Frau Prokop von der Hochschule Bochum für die ausgezeichnete Organisation des Workshops.

Auch das Begleitprogramm war interessant und ansprechend. Am Vorabend des Vortragsprogramms trafen wir uns vor dem Bochumer Hauptbahnhof um 18.00 Uhr zu einer Stadtführung. Zu Fuß ging es durch das Zentrum. Wir erfuhren viel über die Geschichte der Stadt als Teil der ehemaligen Bergbauregion im Ruhrgebiet. Nach dem Spaziergang gab es um 20.00 Uhr ein gemeinsames Abendessen im Restaurant Giuseppe, Gerberstraße 2.

Vom Hauptbahnhof aus war die Hochschule Bochum über die Campuslinie U35 leicht erreichbar. Auch die unmittelbare Nähe zur Ruhr-Universität Bochum ist sicher ein Vorteil für Kooperationen.

Aus Altersgründen werde ich den IngMath-Workshop bald nicht mehr begleiten können. Aus dieser Sicht gab es zwei positive Überraschungen:

1. Prof. Stempel aus Darmstadt erklärte sich bereit, eine Online-Plattform für die Workshop-Reihe einzurichten, auf der Informationen und Proceedings zur Verfügung stehen sollen. Inzwischen steht diese zur Verfügung: <https://fbmn.h-da.de/ingmath>
2. Der Ort für den Workshop 2024 steht fest. Es ist Bielefeld. So bin ich guten Mutes, dass auch nach meinem Ausscheiden der kreative Meinungsaustausch über den Mathematikunterricht in Deutschland weitergeht.

Dieter Schott

Programm des 18. Workshops am 10. 11. 2023

9:00	Claudia Frohn-Schau Begrüßung Dieter Schott Historische Anmerkungen zur Mathematiklehre
9:30	Keynote Speaker Dr. Michael Kallweit Einsatz digitaler Mathematikaufgaben: Innovative Lehransätze mit STACK
10:20	<i>Kaffeepause</i>
10:45	Torsten Stempel Automatische Aufgaben in Moodle mit dem STACK-PlugIn
11:20	Aviva Kaiser Stack Net: Ein Netzwerk für die deutschsprachige Stack-Community
11:55	Angela Schmitz Klassifizierung von STACK-Aufgaben für einen hochschulübergreifenden Aufgabenpool
12:15	<i>Mittagspause</i>
13:15	Patricia Graf, Heiko Knospe, Andreas Schwenk Mathe:Buddy -- eine gamifizierte Lern-App für die Höhere Mathematik
13:50	Hildo Bijl Step-Wise: eine neue Übungshilfe-App für Just-in-Time Learning
14:25	Malte Neugebauer Steigerung der Motivation zum Selbstlernen durch gamifizierte Mathematik – Aufgaben in Lernmanagementsystemen
14:55	<i>Kaffeepause</i>
15:20	Alexander Dominicus Automatische Überprüfung der Anwendung mathematischer Näherungsverfahren durch Coderunner
15:55	Jürgen Vorloeper Symbolisches und Numerisches Rechnen mit (Lua-) LaTeX – Möglichkeiten zur Gestaltung von Lehrmaterial in der Ingenieurmathematik
16:30	Georg Hoever Integration von Mumie-Aufgaben in der höheren Mathematik-Ausbildung
17:00	Claudia Frohn-Schau Verabschiedung und Ausblick

Dieter Schott

Mathematiklehre an Schulen und Universitäten zu Zeiten von Gottlob Frege

Zusammenfassung: Gottlob Frege ist einer der größten Denker aller Zeiten. Sein Mathematikabitur von 1869 ist noch heute zugänglich. In der Mathematik verlangte Frege Klarheit, Exaktheit und logische Schlüssigkeit, die er bei seinen Fachkollegen nicht selten vermisste. Als Mathematiker, Logiker und Philosoph hat er Wesentliches zur Grundlegung der Mathematik beigetragen. Sein Wirken ist auch für die Mathematiklehre der Gegenwart von Bedeutung. Einige Aspekte werden hier genannt.

1. Der Neuhumanismus

Der Neuhumanismus entstand im 19. Jahrhundert als Ausdruck des Rationalismus mit einer Rückbesinnung auf humanistische Ideale. Unter *Wilhelm von Humboldt* wurde in Preußen eine Bildungsreform durchgeführt, die auf ganz Deutschland ausgedehnt wurde und die auf den folgenden, noch heute gültigen Postulaten beruhte:

- Einheit und Freiheit von Lehre und Forschung,
- Einheit der Wissenschaften bei Anerkennung ihrer Eigenständigkeit,
- Zweckfreiheit der Wissenschaften,
- Bildung zur Entwicklung der Persönlichkeit (Bildungsbürger).

Wo Licht ist, da ist bekanntlich auch Schatten. Die Überhebung des Elitären in dieser Zeit durch idealisierte Heldenverehrung und mythische Schwärmerei führt leicht in Abgründe, wie man heute weiß.

Literatur: [1], [2: S. 9-27]

2. Höhere Töchterschulen

Höhere Töchterschulen gab es schon relativ lange. In Königsberg z.B. wurde eine solche 1817 gegründet, aber erst 1908 erlangten die Mädchen dort auch die Hochschulreife. Neben staatlichen fanden sich private Schulen dieser Art. Genehmigungen und Anforderungen wurden regional festgelegt. Im Mathematikunterricht spielten vor allem Rechnen mit Zahlen (Arithmetik) und (Euklidische) Geometrie eine Rolle.

Alexander Frege (1809-1866), der Vater von Gottlob Frege, stellte beim Rat der Stadt Wismar den Antrag, ganz im Sinne des Neuhumanismus eine Höhere Töchterschule einzurichten, um den gestiegenen Bildungsbedarf von Frauen sicherzustellen. Da ein Schulgeld zu entrichten war, konnten nur wohlhabendere

Eltern ihre Töchter dort unterrichten lassen. Ziel war eine angemessene Bildung der Frauen, um den späteren Ehemännern geistig gewachsen zu sein. Nach der Genehmigung 1833 leitete Alexander Frege die Schule bis zu seinem Tod 1866. Danach übernahm seine Frau Auguste Frege (1815-1898) die Schule, die vorher schon als Lehrerin dort tätig war. Der ältere Sohn Gottlob Frege (1848-1925) ging von Oktober 1854 bis Ostern 1869 auf die Große Stadtschule in Wismar, legte dort sein Abitur ab, studierte danach in Jena und Göttingen und begann nach Promotion in Göttingen und Habilitation in Jena eine Universitätslaufbahn. Die Mutter Auguste gab die Höhere Töchterschule 1878 ab und folgte danach ihrem Sohn nach Jena, um ihn zu umsorgen und finanziell zu unterstützen.

Literatur: [6], [3]

3. Die Große Stadtschule in Wismar

Die Große Stadtschule in Wismar hatte einen guten Ruf. Zu Gottlob Freges Schulzeit gab es etwa 450 Schüler, wovon etwa 160 Schüler aus der ländlichen Umgebung kamen. Von den 13 Lehrern waren immerhin 8 promoviert, darunter auch Gottlobs Mathematiklehrer.

Der Rektor Dr. Eduard Haupt (1805-1868) saß nach der bürgerlich-demokratischen Revolution 1848 in der Frankfurter Nationalversammlung. 1864 hielt er zur Eröffnung der Prüfungsperiode an der Schule eine vielbeachtete Rede. Darin sah er die Gefahr, dass die *Jugend* in bisher nicht gekanntem Maße *mit Kultur, Reizen und Genüssen überschüttet* wird. Das Ziel der Schulbildung bestand für ihn in einer *harmonischen Entwicklung der geistigen und sittlichen Anlagen im Sinne der Humanität*. Zur Verstandesentwicklung trügen vor allem die alten Sprachen und die Mathematik bei, die alten Sprachen durch Vermittlung ihrer Grammatik und durch ihren Zugang zur antiken Philosophie, die Mathematik durch Erziehung zum rationalen und begrifflichen Denken.

Literatur: [3]

4. Pädagogische Vorstellungen zum Mathematikunterricht

Beim Mathematikunterricht an der Großen Stadtschule in Wismar wird die *eigenständige* Lösung bestimmter Aufgabentypen aus den Bereichen Planimetrie, Stereometrie, Trigonometrie und Algebra angestrebt. Verlangt wird Sicherheit bei *allgemeinen* algebraischen Umformungen und beim Umgang mit logarithmischen und trigonometrischen Tafeln für numerische Berechnungen. Daneben gibt es durchaus Aufgaben mit speziellen Vorgaben und Anwendungen. Auf Erklärungen und Erläuterungen in Textform wird beim schriftlichen Lösen kein Wert gelegt.

Während ein allgemeiner Funktionsbegriff und die Infinitesimalrechnung noch fehlen, gibt es Aufgaben zur sphärischen Trigonometrie.

Die Pädagogik und Methodik im Fach Mathematik ist regional organisiert. Es gibt in Deutschland dazu durchaus unterschiedliche Vorstellungen, die vom konservativen „Eintrichtern“ von Lösungswegen bis zu kreativen, heuristischen Lösungsstrategien reichen.

Literatur: [5], [7], [9]

5. Das Mathematikabitur von Gottlob Frege 1869

Die schriftliche Abiturprüfung im Fach Mathematik fand im Frühjahr 1869 statt. Mit Gottlob Frege waren 6 Schüler anwesend, von denen 4 bestanden, zwei mit gut und zwei mit ausreichend. Dabei ist zu beachten, dass damals wesentlich strengere Maßstäbe galten als heute. Mit dem Prädikat ‚gut‘ schnitten Gottlob Frege und Heinrich Baumann ab. Während Frege im Anschluss Mathematik studierte und sich in diesem Fach promovierte und habilitierte, war es Baumann bestimmt, später das Rittergut seines Vaters zu übernehmen. So unterschiedlich können die Berufslaufbahnen von ähnlich talentierten jungen Menschen verlaufen.

Es gab 12 handschriftlich vorliegende Aufgaben, von denen 6 gelöst werden sollten. Frege bearbeitete genau 6 Aufgaben bis auf Kleinigkeiten vollständig und in sauberer, gut lesbarer Handschrift. Über die zur Verfügung stehende Zeit ist nichts bekannt. Die Themengebiete waren

- Polynomiale Gleichungen $P(x) = 0$ bis zum 3. Grad (einschließlich komplexer Lösungen), z.B. $x^3 - 3x^2 - 12x - 112 = 0$,
- Gebrochen rationale Gleichungssysteme $F(x, y) = 0$, $G(x, y) = 0$,
- Lineare Gleichungssysteme mit bis zu 3 Unbekannten (Bestimmung ganzzahlige Lösungen),
- Trigonometrie in Dreiecken und Vierecken (Tafeln mit trigonometrischen und logarithmischen Funktionen), z.B. *sind die vier Winkel und der Flächeninhalt eines Sehnenvierecks gesucht, wenn man die vier Seiten kennt.*
- Stereometrie (Volumen, Oberflächen von Zylindern, Kugeln, Kegeln einschließlich Schnittkörpern), z.B. *ist aus dem Inhalt (Volumen) und der Oberfläche eines geraden Kreiskegels dessen Höhe zu bestimmen.*

Die Wahlmöglichkeit bei den Aufgaben lässt vermuten, dass man den Schülern bewusst Freiraum für eigene Entscheidungen einräumte.

Literatur: [10], [3]

6. Das Mathematikabitur in Wismar zu Ostern 1899

Genau 30 Jahre nach dem Abiturjahrgang von Gottlob Frege an der Großen Stadtschule in Wismar gab es dort zum Abitur neben speziellen und

angewandten Aufgaben auch wieder allgemeine Aufgaben mit variablen Größen. Zwei Beispiele folgen:

1. Ein Dreieck [ist] zu zeichnen, von welchem die Differenz zweier Seiten, der von ihnen eingeschlossene Winkel und die Differenz der von der Halbierungslinie dieses Winkels auf der Gegenseite gebildeten Abschnitte gegeben sind ($a - b = d_1$, γ , $u - v = d_2$).
2. Eine Kugel vom Radius r soll durch eine Ebene derart geschnitten werden, daß die Kalotte den m -fachen Flächeninhalt des Schnittkreises hat. Wie groß muss der Abstand der Schnittebene vom Kugelzentrum sein und in welchem Verhältnis steht das Volumen des Kugelsegments zum Volumen der ganzen Kugel?

Insgesamt wurden 7 Aufgaben und 1 Extraaufgabe angeboten, für die 5 Stunden Zeit zur Verfügung standen. Die große Wahlmöglichkeit bei den Aufgaben wie im Jahr 1869 war also nicht mehr vorhanden.

Zu dieser Zeit war Gottlob Frege Mathematikprofessor in Jena und hatte seine bahnbrechenden Arbeiten inzwischen veröffentlicht.

Literatur: [11], [4]

7. Die Lehre zur Studienzeit von Gottlob Frege

Gottlob Frege hat während seines Studiums in Jena verschiedene Lehrpersönlichkeiten kennengelernt. Da ist

- der väterliche und pädagogisch geschickte Ordinarius Prof. *Carl Snell* (1806-1886), der als Mathematiker ein breites Interessengebiet hatte und sein Haus für Gesprächsrunden der Jenaer Gesellschaft öffnete,
- der zugewandte Prof. *Hermann Schaeffer* (1824-1900), der die mathematischen Grundvorlesungen las, der mit pädagogischem Feingefühl an die Fassungskraft der Studenten anknüpfte und daher bei ihnen sehr beliebt war und der in der von ihm gegründeten Mathematischen Gesellschaft in Jena auch Studenten mit Vorträgen zu Wort kommen ließ,
- der fordernde und zunehmend prominente Theoretische Physiker Prof. *Ernst Abbe* (1840-1905), der seinem systematischen und logischen Denkstil in der Lehre folgte und erwartete, dass die Studenten dem gewachsen waren und ihn übernahmen,
- der charismatische Philosoph Prof. *Kuno Fischer* (1824-1907), der die Studenten rhetorisch und fachlich begeistern konnte, der sehr unterhaltsam war, aber auch zur Überheblichkeit neigte.

Auch heute findet man durchaus noch solche Lehrertypen. Der Stil von Abbe ist besonders für leistungsstarke Studenten geeignet, die später eine akademische Laufbahn einschlagen wollen. Der Stil von Schaeffer kommt den

leistungsschwächeren Studenten entgegen. Eine gesunde Mischung ist allgemein förderlich und regt Studenten an, ihren eigenen Stil zu entwickeln.

Unter den Studenten gab es wie heute sowohl die wissbegierigen und erfolgsorientierten als auch die erlebnishungrigen, die das Studium als Pflichtübung absolvierten. Aufgrund der geringen Studentenzahlen in Jena war das Verhältnis der Professoren zu den Studenten durchaus vertraut. Teilweise wurde die Lehre sogar bei den Professoren zu Hause abgehalten.

Literatur: [8], [3]

8. Die Lehre von Gottlob Frege

Gottlob Frege imponierte der anspruchsvolle Lehrstil von Prof. Abbe. An ihm orientierte er sich. Da er aber eher introvertiert war, arbeitete er viel an der Tafel, ohne seine Zuhörer im Blick zu haben. Fragen stellte er kaum, und auch die wenigen anwesenden Studenten hielten sich mit Fragen zurück. Die Vorlesungen zu Freges ‚Begriffsschrift‘, in der die moderne zweiwertige Quantorenlogik in einer zweidimensionalen Schreibweise entwickelt wurde, hatten zudem noch einen schwer verdaulichen Inhalt. So konnten nur ausgesprochen gute Studenten von seiner Lehre profitieren. Studenten wie *Gershom Scholem* und *Rudolph Carnap*, später international anerkannte Wissenschaftler, lobten so auch die gedankliche Tiefe und Klarheit seiner Ausführungen. Allgemein war aber durchaus bekannt, dass Frege der bedeutsamste Wissenschaftler der Jenaer Universität war.

Es wundert nicht, dass Frege wegen seiner geringen Lehrwirksamkeit außer seinem Förderer Abbe unter den Mathematikern kaum Fürsprecher hatte. Abbe wusste vor allem die Leistungen Freges in der Grundlagenforschung zu schätzen. Frege selbst kam mit Philosophen wie *Rudolf Eucken*, *Bruno Bauch* und *Paul Linke* oder dem Altphilologen *Rudolf Hirzel* besser aus. Freges unmittelbarer Vorgesetzter, Ordinarius Prof. *Johannes Thomae*, tolerierte zunächst Freges logisches Programm zur Arithmetik, obwohl er diese im Gegensatz dazu formalistisch begründete und seinen Ansatz für viel erfolgversprechender hielt. Als Thomae aber 1906 die Ansichten Freges in einer „Ferienplauderei über gedankenlose Denker“ in den Jahresberichten der Deutschen Mathematiker-Vereinigung (DMV) kritisierte, schlug der in seiner unnachgiebigen und scharfzüngigen Art zurück. Diese Episode beschädigte das kollegiale Verhältnis der beiden Kollegen ernsthaft. Sie gingen sich künftig aus dem Wege.

Literatur: [8], [4]

9. Die Rolle der Mathematik in der Lehre

Ein Zeitgenosse Freges, der in Göttingen lehrende Prof. *Felix Klein* (1849-1925), machte sich um die Entwicklung der Mathematiklehre besonders verdient. Im Jahre 1891 gründete er den ‚Verein zur Förderung des mathematischen und naturwissenschaftlichen Unterrichts‘. Er setzte sich für eine enge Zusammenarbeit von Vertretern der Universitäten und der höheren Schulen ein. Ab 1894 organisierte er eine Weiterbildung von Mathematiklehrern in den Schulferien, um einen modernen anspruchsvollen und zugleich praxisorientierten Unterricht zu garantieren. Im Jahre 1908 wurde der ‚Deutsche Ausschuss für mathematischen und naturwissenschaftlichen Unterricht‘ (DAMNU) gegründet. In den Jahren von 1909 bis 1916 verfasste Felix Klein Abhandlungen über den mathematischen Unterricht in Deutschland.

Literatur: [7]

10. Ein Fazit

Gerade in letzter Zeit wird wieder viel über den Wert des Mathematikunterrichtes diskutiert. Schüler und Studenten haben zunehmend Schwierigkeiten, die Anforderungen zu erfüllen. Wochenstunden werden im Fach gestrichen, Lehrpläne gekürzt. Von mir kommt da ein klares Plädoyer für eine Stärkung der Mathematik als wesentliche Bildungsgrundlage. Die Beschäftigung mit der Mathematik

- erzieht zum objektiven, logischen, abstrakten und wissenschaftlichen Denken und Argumentieren,
- bewirkt generell Präzision in der Sprache,
- erleichtert das Verständnis moderner wissenschaftlicher und technischer Entwicklungen,
- schützt vor unzulässigen Verallgemeinerungen und Fehlschlüssen,
- vermindert die Gefahr von Missverständnisse in Streitgesprächen.

Mathematik sollte daher ein wesentlicher Bestandteil von Bildung bleiben. Natürlich müssen moderne Hilfsmittel genutzt werden, um den Mathematikunterricht zeitgemäß und attraktiv zu gestalten. Die Lehrkräfte der Mathematik sind für den guten Ruf der Mathematik zuständig. Nehmen Sie Ihre Verantwortung wahr und leisten Sie Ihren Beitrag für eine Stärkung der Mathematik im allgemeinen Bildungskanon!

Literatur: [9]

Literaturverzeichnis

1. Bernd, Heinz-Helmut: Hauptfach Mathematik. Über Neuhumanismus, Wertewandel und heutige Befindlichkeiten. Gottlob Frege – Bildungsbürger im Systemwandel. Wismarer Frege-Reihe, Heft 01/2022.
2. Elstroth, Jürgen; Schmitz, Norbert: Geschichte der Mathematik an der Universität Münster, Teil I: Zeitraum 1773-1945. Münster 2008. <http://www.math.uni-muenster.de/historie>
3. Framm, Edith; Framm, Joachim; Schott, Dieter: Gottlob Frege. Der Aristoteles aus Mecklenburg. Romanbiografie – Erster Teil – Von der Kindheit und Jugend bis zur Begriffsschrift. Zweite bearbeitete Auflage. Verlag Koch und Raum, Wismar 2022.
4. Framm, Edith; Framm, Joachim; Schott, Dieter: Gottlob Frege. Der Aristoteles aus Mecklenburg. Romanbiografie – Zweiter Teil – Professorenzeit in Jena und Rückkehr in die Heimat. Erste Auflage. Verlag Koch und Raum, Wismar 2024.
5. Grube, August Wilhelm: Leitfaden für das Rechnen in der Elementarschule nach den Grundsätzen einer heuristischen Methode. Berlin 1881.
6. Hinz, Margund: Das höhere Schulwesen der Stadt Königsberg in Preußen 1800-1915. Duncker & Humblot, Berlin 2021.
7. Klein, Felix: Elementarmathematik vom höheren Standpunkt aus (1907-08). Nachdruck Springer, Berlin u.a. 1968.
8. Kreiser, Lothar: Gottlob Frege. Leben – Werk – Zeit. Meiner Verlag, Hamburg 2001.
9. Meschkowski, Herbert: Mathematik als Grundlage. Ein Plädoyer für ein rationales Bildungskonzept. Deutscher Taschenbuch-Verlag 1973.
10. Schott, Dieter: Das Mathematikabitur von Gottlob Frege. Wismarer Frege-Reihe, Heft 01/2020.
11. Schott, Dieter: Das Mathematikabitur in Wismar zu Ostern 1899. Wismarer Frege-Reihe, Heft 01/2022.

Autor

Prof. Dr. rer. nat. habil. Dieter Schott

Fakultät für Ingenieurwissenschaften, Gottlob-Frege-Zentrum

Hochschule Wismar

Philipp-Müller-Str. 14

D-23966 Wismar

E-Mail: dieter.schott@hs-wismar.de

Jürgen Vorloeper

Symbolisches und Numerisches Rechnen mit Lua \LaTeX

Zusammenfassung. Die in Lua \LaTeX integrierte Skriptsprache Lua ermöglicht es, während eines \LaTeX -Durchlaufs Python-Code auszuführen und die Ergebnisse direkt im \LaTeX -Dokument weiterzuverarbeiten. So können in einem \LaTeX -Durchlauf anspruchsvolle numerische und symbolische Berechnungen durchgeführt und die Ergebnisse in Form von Tabellen, Grafiken und Textelementen aufbereitet werden. Dies eröffnet vielfältige Möglichkeiten bei der Erstellung von Lehrmaterialien im technisch-naturwissenschaftlichen Bereich.

Zielsetzung

Im mathematischen Grundlagenbereich werden den Studierenden regelmäßig standardisierte Übungsaufgaben zur selbstständigen Arbeit gestellt. In jüngerer Zeit erfolgt dies häufig digitalisiert über Lernplattformen. Gleichwohl sind klassische Übungsaufgaben auf Papier oder als pdf-Datei im Hochschulalltag nach wie vor weit verbreitet. Für den Textsatz im mathematisch-naturwissenschaftlichen Bereich ist \LaTeX in den vergangenen vier Jahrzehnten zu einem Standard geworden. In jedem Fall ist die Erstellung von mathematischen Übungsaufgaben für die Lehrenden mit viel Aufwand verbunden, und die Lerngruppen wünschen sich häufig weiteres Übungsmaterial. Daher ist es wünschenswert, den zeitlichen Arbeitsaufwand zur Erstellung von Lernmaterialien zu reduzieren. Diese Zielsetzung hat beispielsweise das Projekt MATeX [3], das mit Matlab einen sehr mächtigen technischen Unterbau hat. In den vergangenen Jahren ist auch die Entwicklung im \LaTeX -Umfeld weiter gegangen, und es stehen inzwischen verschiedene Möglichkeiten zur Verfügung, \LaTeX mit externen Programmiersprachen zu kombinieren. Diese können förderlich bei der Erstellung von Lehrmaterialien im technisch-naturwissenschaftlichen Bereich eingesetzt werden.

Hintergrund

Der Begriff \LaTeX umfasst eine Vielzahl an Werkzeugen und Paketen, die auf dem ursprünglich von Donald Knuth entwickelten Satzsystem \TeX basieren und dessen ursprüngliche Funktionalität erheblich erweitern. Dieses Satzsystem wird überwiegend im technisch-naturwissenschaftlichen Bereich eingesetzt und ist dort zu einem Standard geworden. Eine der modernsten \LaTeX -Engines ist $\text{Lua}\TeX$, die heutzutage in jeder der großen \LaTeX -Distribution standardmäßig enthalten ist. Unter den Begriff $\text{Lua}\TeX$ fassen wir im Folgenden die $\text{Lua}\TeX$ -Engine mitsamt der zugehörigen Makro-Pakete zusammen.

$\text{Lua}\TeX$ ist eine Weiterentwicklung von früheren \LaTeX -Engines. Sie bietet Unterstützung für Unicode, ermöglicht Zugriff auf Schriften im OpenType-Format und integriert METAPOST . Wesentliches Merkmal von $\text{Lua}\TeX$ ist die darin integrierte Skriptsprache Lua. Diese ist eine sehr kompakte und (weitgehend) plattformunabhängige Skriptsprache, die einfach zu nutzen ist. Sie beinhaltet u. a. ein math-Modul mit Basisfunktionalitäten und stellt eine C-Schnittstelle bereit.

Lua und C-Programme

In $\text{Lua}\TeX$ kann Lua-Programmcode während eines \LaTeX -Durchlaufs ausgeführt werden. Damit lassen sich mathematische Routinen recht einfach in ein \LaTeX -Dokument einbinden, s. [4, 7]. Allerdings stellt das math-Modul in Lua nur Basisfunktionalitäten bereit, so dass darüber hinaus gehende Verfahren selbst geschrieben werden müssen. Der Vorteil der Plattformunabhängigkeit von Lua steht hier die eingeschränkte Verfügbarkeit mathematischer Module entgegen.

Über die C-Schnittstelle von Lua lassen sich Funktionen aus externen C-Bibliotheken aufrufen. Insbesondere für numerische Methoden stehen zahlreiche exzellente und frei verfügbare C-Bibliotheken zur Verfügung. Damit erweitert sich das Anwendungsfeld von numerischen Berechnungen in einem Latex-Dokument enorm, s. [6]. Geeignete C-Bibliotheken für symbolische Berechnungen mit einem breiten Anwendungsfeld sind dagegen kaum verfügbar, s. [2]. Die C-Schnittstelle in Lua baut elementar auf einem Stack auf, so dass zu ihrer Nutzung Low-Level-Programmierung

erforderlich ist. Zugleich bietet Lua nur eine rudimentäre Debugging-Funktionalität. In der Praxis sind der Verwendung von C-Bibliotheken daher Grenzen gesetzt.

Python-Programme

Im Bereich des wissenschaftlichen Rechnens gewinnen seit einigen Jahren Sprachen wie Python oder Julia an Bedeutung. Diese sind einfach zu erlernen, stellen umfangreiche und qualitativ exzellente mathematische Pakete für numerische und symbolische Berechnungen bereit und sind auf vielen Plattformen verfügbar.

Mit dem recht neuen Paket *pyluatex* [1] lässt sich mit wenig Aufwand Python-Code in Lua \LaTeX verwenden. Dies eröffnet die Möglichkeit, die oben aufgezeigten praktischen Beschränkungen zu umgehen und mathematische Funktionalitäten direkt in ein \LaTeX -Dokument einzufügen. Im Unterschied zu ähnlichen Paketen wie *pythontex* [5] ist bei Verwendung von *pyluatex* nur *ein* \LaTeX -Durchlauf erforderlich.

Das folgende Beispiel möge einen ersten Eindruck verschaffen: Sofern Python nicht im aktuellen Suchpfad verfügbar ist, muss der Pfad beim Einbinden des Pakets *pyluatex* angegeben werden. Sodann kann im \LaTeX -Dokument in einer `python`-Umgebung Python-Code eingegeben werden. Darin kann man Python-Pakete importieren, Strings verarbeiten und natürlich auch Berechnungen durchführen.

```

1 \usepackage[executable=/usr/bin/python3]{pyluatex}
2
3 \begin{python}
4 import math
5 import random
6 random.seed(0)
7 from scipy.stats import norm
8
9 text = 'Hallo Hochschule Bochum!'
10 \end{python}
11
12 \begin{itemize}
13   \item \py{text}
14   \item Wurzel Zwei ist  $\sqrt{2} = \text{py}\{\text{math.sqrt}(2)\}$ 
15   \item Eine Zufallszahl:  $\text{py}\{\text{random.randint}(2,5)\}$ 
16   \item Standardnormalverteilung  $\Phi(1.26) \approx \text{py}\{\text{round}(\text{norm.cdf}(1.26),5)\}$ 
17 \end{itemize}

```

Der Befehl `\py` leitet die Python-Ausgabe an das \LaTeX -Dokument weiter. Da mit Python ein externes Programm ausgeführt wird, ist die Option `-shell-escape` beim Befehl `lualatex` erforderlich. Insgesamt ist die Verwendung des Pakets `pyluatex` recht einfach. Es gibt nur wenige Paketoptionen, und es ist nur *ein* \LaTeX -Durchlauf erforderlich.

Beispiel

Das folgende Beispiel zeigt die Verwendung von Python-Paketen zum Erstellen einer randomisierten Aufgabe mit Lösung aus dem Grundlagenbereich. In dem folgenden Python-Code wird eine randomisierte 4×4 -Matrix \mathbf{A} mit einem Parameter a erstellt. In Abhängigkeit von a wird symbolisch $\det(\mathbf{A})$ berechnet und schließlich die Gleichung $\det(\mathbf{A}) = 0$ symbolisch gelöst. Der Zufallszahlengenerator wird wie üblich über einen Seed gesteuert. Die Größen, die eine \LaTeX -Formatierung benötigen, werden mittels der \LaTeX -Ausgabe von Sympy als Zeichenkette gespeichert.

```

1 \begin{python}
2     import sympy as sp
3     from sympy.matrices import Matrix
4     import random
5
6     random.seed(1)
7
8     a = sp.symbols("a")
9     a_str = sp.latex(a)
10
11     A = Matrix([[2, 1, 0, -1], [0, 1, 4, 1], [2, 2, 0,
12                 1], [0, 2, 0, a]])
13     A[1, 2] = random.randint(1, 6)
14
15     A_str = sp.latex(A)
16     A_str = A_str[6:-7]
17     A_str = A_str.replace("matrix", "pmatrix")
18
19     detA_str = sp.latex(A.det())
20
21     eqn = sp.Eq(A.det(), 0)
22     x = sp.solve(eqn, a, dict=False)
23     x_str = sp.latex(x[0])
24 \end{python}

```

Die in Python berechneten Größen können nun im \LaTeX -Dokument mittels `\py` verwendet werden. So wird zunächst der kurze Aufgabentext

erstellt und anschließend die Lösung erzeugt.

```

25 Aufgabe: Gegeben ist die Matrix
26 \begin{equation*}
27 \mathbf{A} = \py{A_str}
28 \end{equation*}
29 mit  $\py{a_str} \in \mathbb{R}$ .
30 \begin{enumerate}
31 \item Berechnen Sie  $\det(\mathbf{A})$ .
32 \item Für welche  $\py{a_str} \in \mathbb{R}$  ist  $\mathbf{A}$ 
    eindeutig invertierbar?
33 \end{enumerate}
34
35 Lösung: Es ist  $\det(\mathbf{A}) = \py{detA_str}$ , und
     $\mathbf{A}$  ist nur für  $\py{a_str} \neq \py{x_str}$ 
    eindeutig invertierbar.

```

Anwendungsfeld

Die in den vorangegangenen Abschnitten beschriebene Vorgehensweise bietet sich vor allem zur Erstellung von Lehrmaterialien mit standardisierten Aufgaben an. Es lassen sich auf diese Weise unterschiedliche Aufgabentypen parametrisieren. Diese können für individualisierte Übungsaufgaben eingesetzt werden, auch in Kombination mit studentischen Peer-Review Lehrformen. Generell ist es sinnvoll, einen Pool an parametrisierten Aufgaben zu unterschiedlichen Themen aufzubauen. Ein weiteres Einsatzgebiet ist die automatisierte Erstellung von Tabellen. Hierzu zählen beispielsweise Tabellen von Ableitungs- und Stammfunktionen, Quadraturformeln, Verteilungsfunktionen, um nur einige bereits realisierte Beispiele zu nennen. Bei umfangreichen Rechnungen lassen sich leicht, schnell und fehlerfrei (Zwischen-) Ergebnisse in ein \LaTeX -Dokument einfügen.

Zwingend erforderlich ist die Verwendung einer aktuellen \LuaTeX -Engine. Viele Verlage oder auch arXiv setzen die Benutzung älterer Engines voraus, etwa \pdfTeX . Daher ist der Einsatz der vorgestellten Möglichkeiten bei Forschungspublikationen oder bei Buchprojekten in der Regel nicht möglich.

Technische Entwicklungen im \LaTeX -Bereich erfolgen oftmals langsam. Dafür werden sie in der Regel über einen langen Zeitraum gepflegt und sind frei verfügbar. Auch Python und Lua werden aktiv weiterentwickelt. Daher ist zu erwarten, dass die hier vorgestellten Möglichkeiten

auch zukünftig zur Verfügung stehen werden. Der technische Unterbau ist sehr kompakt gehalten, und es besteht keine Abhängigkeit zu proprietärer Software von Drittanbietern.

Literaturverzeichnis

- [1] **Enderle, T.:** *pyluatex*, <https://ctan.org/pkg/pyluatex>.
- [2] **Flinkow, T.:** *A Framework for Individualised Mathematical Assignments with Solutions in L^AT_EX*, Bachelorarbeit. Hochschule Ruhr West, (2000).
- [3] **Helfrich-Scharbanenko, A.; Kapedius, K.; Rutka, V.; Sommer, A.:** *Mathematische Aufgaben und Lösungen automatisch generieren*. Springer, (2018).
- [4] **Montijano, J.I.; Pérez, M.; Rández, L.; Varona, J.L.** *Numerical methods with LuaL^AT_EX*. TUGboat 35.1 (2014).
- [5] **Poore, G.M.:** *pythontex*, <https://www.ctan.org/pkg/pythontex>.
- [6] **Vorloeper, J.:** *Numerisches Rechnen mit LuaL^AT_EX*, https://github.com/dante-ev/Vortraege_Tagungen/blob/master/2021-Fruhjahr/Vorloeper_RechnenLuaTeX.pdf
- [7] **Voss, H.:** *Chaotische Symmetrien mit Lua berechnen*. Die T_EXnische Komödie. 32.3, (2020).

Autor

Prof. Dr. rer. nat. Jürgen Vorloeper
Institut Naturwissenschaften
Hochschule Ruhr West
Duisburger Straße 100
D-45479 Mülheim an der Ruhr
E-Mail: juergen.vorloeper@hs-ruhrwest.de

Alexander Dominicus

Automatisiertes Assessment iterativer Verfahren im Mathematikbereich in Moodle

Zusammenfassung. Dieser Artikel beschreibt den Einsatz des Lern-Management-Systems Moodle in Kombination mit dem Plugin *Coderunner* beim Erlernen iterativer mathematischer Verfahren in der Hochschullehre. Es wird dargestellt, wie Coderunner Studierenden in Fächern wie Mathematik, Ingenieurwissenschaften und Informatik hilft durch direktes und präzises Feedback praktische Fähigkeiten im Umgang mit iterativen Algorithmen zu entwickeln. Insbesondere wird beschrieben wie Bewertungen individualisiert erfolgen können, um Lösungen iterativer Algorithmen qualitativ beurteilen zu können.

Einleitung

Im akademischen Kontext, insbesondere in Fächern wie Mathematik, Ingenieurwissenschaften und Informatik spielen iterative Verfahren eine zentrale Rolle. Diese Methoden, darunter das näherungsweise Lösen von Gleichungen, die Darstellung von Zahlen, die Lösung von Differentialgleichungen oder die Approximation von Funktionen durch Polynome, sind wesentliche Werkzeuge in der Hochschullehre. Der Erwerb von Fähigkeiten im Umgang mit diesen Verfahren ist essentiell, erfordert jedoch auch regelmäßige praktische Anwendungen, um ein tiefgreifendes Verständnis zu entwickeln. In den meisten Fällen verlangt dies eine intensive und zeitaufwändige Betreuung durch die Dozierenden: Es müssen Fragestellungen zur Bearbeitung durch die Studierenden entwickelt werden. Daraus resultiert ein studentischer Programmcode, der in der Regel manuell von den Dozierenden ausgeführt und bewertet wird. Motiviert durch [6] soll diese Form des Assessments vollständig automatisiert werden. Durch automatisiertes Assessment kann die Bewertung und Benotung erheblich vereinfacht und objektiver werden. Zudem wird entsprechendes Feedback sofort ausgegeben und motiviert die Studierenden gegebene Fragestellungen vollständig und korrekt zu bearbeiten ([8]). Eine Schlüsselrolle beim automatisierten Assessment spielt der Fragetyp Coderunner [7, 1] für das Moodle-System [2]. Coderunner ermöglicht es den Studierenden, einen Programmcode direkt im Moodle-System einzugeben und zu testen.

Coderunner

Coderunner ist ein innovativer Fragetyp innerhalb des Moodle-Systems, der speziell für die automatische Überprüfung von Programmcode konzipiert ist. Er wird in Moodle-Tests eingesetzt, um Studierenden eine praktische Programmiererfahrung zu ermöglichen. Eine Besonderheit von Coderunner ist die Unterstützung einer Vielzahl von Programmiersprachen, darunter Python3, C, C++, Java, PHP, JavaScript (NodeJS), SQL und Octave/Matlab. Der Einsatz des Fragetyps Coderunner läuft im einfachsten Fall folgendermaßen ab: Anhand der Fragestellung erstellen die Studierenden eine Funktion. Diese wird auf einem externen Server mit bestimmten, im Vorfeld definierten, Parametern (*Testfälle*) ausgeführt und die Ausgaben des Programmcodes an das Moodle-System zurückgegeben. Die erhaltenen Ausgaben werden mit den erwarteten Ausgaben der Testfälle verglichen und bewertet. Stimmen alle Ausgaben der studentischen Funktion mit den hinterlegten exakt überein, wird die Frage als "korrekt" gewertet. Andernfalls wird die Frage als "falsch" gewertet und die Studierenden müssen ihre Funktion entsprechend anpassen. Ein praktisches Beispiel wäre die folgende Python-Funktion, welche für eine Zahl n dessen Quadratzahl n^2 ausgibt:

```
def printsquare(n):
    print(n*n)
```

Diese Funktion lässt sich durch Testfälle überprüfen (`printsquare(5)`, `printsquare(-3)`, `printsquare(0)` usw.) und damit auch bewerten. Für die Bewertung iterativer mathematischer Näherungsverfahren, wie sie in diesem Artikel diskutiert werden, reicht der Vergleich auf exakte Übereinstimmung jedoch nicht aus. Dies lässt sich an folgender Musteraufgabe verdeutlichen:

Musteraufgabe. Erstellen Sie eine Python-Funktion `appsqrt(n,x)`, die auf Basis des Newton-Verfahrens eine Approximation an die Quadratwurzel der Zahl n berechnet. Als Startwert x_0 der Iteration wird die Zahl n gewählt. Die Funktion soll so erstellt werden, sodass für $\ell \geq 1$ und

$x_\ell := \text{appsqrt}(n, \text{appsqrt}(n, x_{\ell-1}))$ mit $x_0 := \text{appsqrt}(n, n)$ gilt:

$$\begin{aligned}
 |\text{appsqrt}(n, x_0) - \sqrt{n}| &= |\text{appsqrt}(n, n) - \sqrt{n}| \\
 &\geq |\text{appsqrt}(n, x_1) - \sqrt{n}| \\
 &\geq |\text{appsqrt}(n, x_m) - \sqrt{n}| \\
 &\geq |\text{appsqrt}(n, x_{m+1}) - \sqrt{n}|
 \end{aligned} \tag{1}$$

für alle $m \geq 1$. D.h. durch iterierte Funktionsanwendung wird die Genauigkeit der Berechnung verbessert.

Die folgende Funktion beschreibt eine mögliche Lösung der Musteraufgabe:

```
def appsqrt(n, x):
    new_approx = 0.5 * (x + n/x)
    return new_approx
```

In diesem Fall besteht die Aufgabe nicht darin eine bestimmte Ausgabe zu erzeugen, die mit einer hinterlegten Lösung verglichen wird. Stattdessen soll der studentische Programmcode qualitativ im Hinblick auf (1) getestet werden. Um diese Aufgabe im Moodle-System darzustellen, muss eine eigene Bewertungsvorlage einer Coderunner-Frage erstellt werden.

Eigene Bewertungsvorlagen in Coderunner entwickeln

Grundlage der Bewertung einer Coderunner-Frage ist die hinterlegte Bewertungsvorlage. Diese ist im hohen Maße adaptierbar, wodurch eine präzise Beurteilung des studentischen Programmcodes sichergestellt werden kann. Dies wollen wir hier genutzt, um eine qualitative Bewertung der Musteraufgabe auf Basis von (1) sicherzustellen. Um diese Anpassung vorzunehmen, navigiert man im Moodle-System zum Bereich "Frage bearbeiten". Unter dem Reiter *Anpassung* findet man die Option *Anpassen*, welche den Zugang zu weiteren Einstellungen ermöglicht. Nach der Auswahl der Option *Anpassen* öffnet sich ein Editor, in dem ein selbst erstelltes Python-Skript hinterlegt werden kann, das für die Bewertung und Benotung der eingereichten Lösungen verwendet wird. Dies lässt eine flexible und präzise Beurteilung zu, da das Bewertungsskript speziell auf die Anforderungen der jeweiligen Aufgabe zugeschnitten werden kann. Auf diese

Weise lässt sich sicherstellen, dass die Bewertung nicht nur auf der Grundlage einer einfachen Übereinstimmung erfolgt, sondern auch komplexere Aspekte der Programmlösung berücksichtigt werden können. Im Fall der Musteraufgabe wird die folgende Bewertungsvorlage erstellt:

```

import math
import random
from random import uniform
NUM_ITER = 16
{{ STUDENT_ANSWER | e('py') }}
ok = True
TESTS=21
TOLERANCE=10e-6
for test in range(-1,TESTS):
    if test > TESTS:
        break
    n = random.randint(1, 10000)
    expected = math.sqrt(n)
    startvalue=n
    lasterror=abs(startvalue-expected)
    for i in range(NUM_ITER):
        stud_answer = appsqrt(n, startvalue)
        startvalue = stud_answer
        newerror=abs(stud_answer-expected)
        if lasterror<newerror:
            print("Der Algorithmus nähert sich nicht der exakten Lösung an!")
            ok = False
            break
        lasterror = newerror
    if abs(expected - stud_answer) <= TOLERANCE:
        print("Gewünschte Toleranz erreicht!")
        print("n=",n)
        print("Anzahl Iterationen: {}, Fehler: {}".format(i, newerror))
    else:
        print("Gewünschte Toleranz nicht erreicht!")
        print("Fehler= ", newerror)
        ok = False
        break
if ok:
    print("Alle Tests bestanden!")

```

Um die Lesbarkeit zu erhöhen wurde das obige Template gekürzt. Das vollständige Python-Skript befindet sich im Aufgabenpool des DigiTeach-Instituts [3]. Neben dieser Aufgabe sind dort auch weitere Beispiele zur individuellen Bewertung vorhanden.

Im obigen Bewertungsskript werden als erstes alle erforderlichen Pakete importiert, die studentische Lösung mit Hilfe der Twig Template-Engine ausgelesen sowie alle benötigten Konstanten gesetzt. Daraufhin beginnt die erste Schleife, welche eine zuvor festgelegte Anzahl an Testfällen durchläuft (20 Testfälle): Es werden eine zufällige natürliche Zahl zwischen 1

und 10000 gewählt, die entsprechende Wurzel als Referenzlösung berechnet, der Startwert gesetzt und der Anfangsfehler berechnet. Anschließend wird in einer weiteren Schleife das Newton-Verfahren (die studentische Funktion) gemäß der vorgegebenen Anzahl an Iterationen durchgeführt. Dabei wird bei jedem Iterationsschritt geprüft, ob die neue Lösung näher (verglichen mit der Lösung aus dem vorangegangenen Iterationsschritt) an der Referenzlösung liegt. Falls der Fehler sich vergrößert hat, wird die Bewertung abgebrochen und die studentische Lösung nicht gewertet. Ist der Fehler im Iterationsschritt kleiner geworden, wird die nächste Iteration durchgeführt. Sind alle Iterationsschritte durchlaufen, wird geprüft, ob die Newton-Iteration unter einer vorgegebenen Schranke liegt. Wenn ja, wird den Studierenden ein Feedback zum Testfall ausgegeben: Die Testzahl n , die Zahl der Iterationen und der verbleibender Fehler. Am Ende wird den Studierenden ein finales Feedback zum erfolgreichen Abschluss der Testfälle ausgegeben.

Ausblick

Diese Arbeit beschreibt den Einsatz automatisierter Assessments im Mathematikbereich. Analog zu [6] sollte eine Untersuchung zur Verbesserung der Studienleistung und/oder Verbesserung der Lernmotivation durchgeführt werden. Eine weitere Einsatzmöglichkeit von Coderunner ist das Schaffen freier Lernumgebungen für Studierende innerhalb des Moodle-Systems. Möglich wird dies durch Nutzung des angepassten Fragetyps *Sandpit* ([4]), welcher nicht auf einer Bewertung des studentischen Programmcodes basiert, sondern lediglich zu dessen Ausführung dient. Somit können Dozierende Lernumgebungen vergleichbar mit einem Jupyter-Notebook innerhalb des Moodle-Systems für die Studierenden bereitstellen. Ein weiterer Schritt wäre dort die Bereitstellung von spezialisierten Software-Tools. Als Beispiel sind hier Finite-Elemente-Toolboxen (vgl. [5]) zu nennen, welche direkt im Moodle-System von den Studierenden verwendet werden könnten. Auf diese Weise können Studierende auf eine interaktive und nutzerfreundliche Art mit solchen Methoden vertraut gemacht werden. Die Installation spezieller Software durch die Studierenden entfällt somit, da sich in diesem Fall die Testumgebungen vollständig im Moodle-System befinden. Daher trägt CodeRunner nicht nur zur Verbesserung der Programmierfähigkeiten bei, sondern erweitert auch das Verständnis für komplexe mathematische Methoden und deren Anwendung.

Literatur

- [1] URL: <https://coderunner.org.nz/>.
- [2] URL: <https://moodle.org/>.
- [3] URL: <https://www.hochschule-bochum.de/digiteach/materialien-downloads/>.
- [4] URL: <https://coderunner.org.nz/mod/page/view.php?id=558>.
- [5] URL: <https://github.com/kinnala/scikit-fem>.
- [6] Florian Horn, Daniel Schiffner und Detlef Krömker. »Akzeptanz der Nutzung von automatisiertem Assessment im Rahmen einer virtuellen Vorlesung«. In: *Proceedings of the Fifth Workshop 'Automatische Bewertung von Programmieraufgaben' (ABP 2021), virtual event, October 28-29, 2021*. 2021.
- [7] Richard Lobb und Jenny Harlow. »Coderunner: A tool for assessing computer programming skills«. In: *ACM Inroads* 7.1 (2016), S. 47–51.
- [8] Zahid Ullah u. a. »The effect of automatic assessment on novice programming: Strengths and limitations of existing systems«. In: *Computer Applications in Engineering Education* 26.6 (2018), S. 2328–2341.

Autor

Dr. Alexander Dominicus
DigiTeach-Institut
Hochschule Bochum
Am Hochschulcampus 1
D-44801 Bochum
E-Mail: alexander.dominicus@hs-bochum.de

Torsten-Karl Stempel

Automatische Aufgaben in Moodle

Zusammenfassung: Übungsaufgaben bereit zu stellen ist wichtig, aber aufwändig. Noch aufwändiger ist die Korrektur. Schließlich „gehen“ Aufgaben und Lösungen „rum“ und es kommt nicht mehr zur inhaltlichen Auseinandersetzung mit der Aufgabe, sondern die Lösungen werden nachvollzogen – „gelesen“. In Moodle kann man verschiedene Aufgabentypen nutzen, die zum einen automatisch eine Rückmeldung zur Richtigkeit erzeugen und zum anderen durch Parameter so variiert werden können, dass jeder Studierende eine eigene Rechnung durchführen kann. Man kann Aufgaben in einem Fragenkatalog sammeln und kategorisieren und so automatisch Aufgabenblätter erzeugen.

1. Einführung

Gute Übungsaufgaben zu erstellen, kostet Zeit und die Korrektur handschriftlicher Abgaben ebenfalls. Durch Wiederholer oder auch auf Webseiten werden Lösungen gesammelt und weitergegeben, so dass nachfolgende Studierende nicht mehr zum Kern der Übung kommen: Erlernte Lösungsverfahren selbstständig auszuprobieren und anzuwenden. Vielmehr werden die Lösungen der Aufgaben nachvollzogen (durchgelesen), was zwar ein gewisses Verständnis fördert, aber nicht die Fähigkeit, bekannte Werkzeuge und Methoden auszuwählen und erfolgreich anzuwenden. Auch wird der Eindruck erweckt, dass man Lösungen einfach findet und dass es ggf. nur einen Lösungsweg gibt.

Eine Möglichkeit, die Problematik abzumildern ist, die Aufgaben mit Zufallszahlen zu parametrisieren, so dass auch wirklich eine Rechnung durchgeführt werden kann. Auf diese Weise müssen die Studierenden die Lösungswege diskutieren, wenn sie ihre Ergebnisse vergleichen wollen.

Eine weitere Erleichterung erhält man als Lehrender durch eine automatische Korrektur der Aufgaben. Hierzu wäre allerdings zum einen eine leistungsfähige Schrifterkennung notwendig, die für die vorliegenden Schriftproben meines Wissens nach noch nicht existiert. Zum anderen benötigt man ein Expertensystem, das die Äquivalenz mathematischer Argumentation erkennt, was ebenfalls für diesen Anwendungsfall noch nicht realisiert wurde. Um dennoch eine Erleichterung der Korrektur zu erhalten, kann man die Lösungen verschieden ausführlich abfragen.

Bezogen auf schriftliche Rechnungen (also keine Zeichnungen und geometrischen Beweise) ist es am einfachsten, Multiple Choice Abfragen zu erstellen. Weiter kann man zahlenmäßige Eingaben (auch mit Einheiten) verlangen bis hin zu algebraischen Ausdrücken, die dann von einem Computer Algebra System (CAS) geprüft werden. Schließlich ist es noch möglich, auch Zwischenschritte einer Rechnung abzufragen, was allerdings eine umfangreiche Aufgabenprogrammierung erfordert.

In der Lernplattform Moodle hat man die Möglichkeit, Aufgaben in standardisierten Datenformaten zu erstellen, in verschiedenen Veranstaltungen mitzuverwenden und mit anderen Lehrenden zu teilen. Dadurch verringert sich (theoretisch) der Aufwand des Einzelnen.

Ein weiterer Vorteil der Lernplattform ist, dass eine Buchführung für die erreichten Punktzahlen vorhanden ist, so dass man Testate am Semesterende einfacher vergeben kann. Auch haben die Studierenden jederzeit einen Überblick über ihren Leistungsstand.

Durch die Corona-Pandemie Anfang 2020 war der Einsatz von Lernplattformen fast „alternativlos“ und die Verwendung automatischer Korrektur eine große Arbeitserleichterung. Moodle bietet auch die Möglichkeit, von den Studierenden eingescannte und hochgeladene Lösungen online zu korrigieren, Feedback zu geben und manuell Punkte zu erfassen. Dies ist zum Beispiel wichtig für Themen oder Aufgabentypen, die nicht oder nur schwer durch einfache Eingabefelder abbildbar sind.

2. Parametrisierung von Aufgaben

Auch ohne Moodle und vor der Corona-Pandemie habe ich parametrisierte Aufgaben eingesetzt. Führt die Parametrisierung „nur“ zu verschiedenen Rechenergebnissen, dann liegt eine Personalisierung der Aufgabe vor. Moodle kann jedem Studierenden mit einem Zufallsgenerator verschiedene Parameter erzeugen, so dass die notwendigen Berechnungen verschiedene Resultate liefern.

Nutzt man evtl. auf Basis der erlangten Punktzahlen Aufgaben mit verschiedenem Schwierigkeitsgrad, dann kann man Studierende verschiedener Leistungsstufen ansprechen, eine Art Binnendifferenzierung.

3. Offline-Einsatz vor Corona

Vor Corona wurde die Parametrisierung durch die Matrikelnummer realisiert. Die nachfolgende Abbildung zeigt eine Aufgabe (Physik für Elektrotechnik) in der eine physikalische Größe von der Matrikelnummer abhängt und so zehn verschiedene Ergebnisse erzeugt werden.

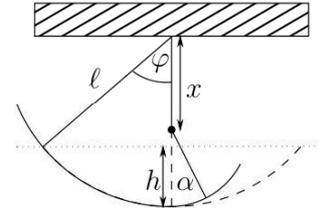
m_1, \dots, m_6 bezeichnen die Ziffern Ihrer Matrikelnummer M .

$M_i = m_i + 1$ ist jeweils um 1 erhöht.

$M^{[p]} := M \bmod p$ ist Rest der Division von M durch p

Aufgabe K04 : (Hemmpendel)

Beim sog. **Hemmpendel** befindet sich ein Hindernis im Abstand x senkrecht unter der oberen Aufhängung. Die Gesamtschwingungsdauer ist die Summe der halben ungehemmten Schwingung mit Pendellänge ℓ und der halben Schwingung mit einer Pendellänge von $\ell - x$



$$T = \frac{T_\ell}{2} + \frac{T_{\ell-x}}{2}$$

- Bestimmen Sie die Schwingungsdauer für das ungehemmte Pendel für $\ell = 0,8$ m. (2)
- Welche Gesamtschwingungsdauer folgt für $x = \ell \cdot M_6/20$? (2)
- Wie groß ist der Winkel α für $\varphi = \frac{\pi}{4}$? (3)
- Wie groß kann der Winkel φ „maximal“ werden? (3)

Zur Korrektur wurden alle Zahlenwerte mit Excel vorberechnet.

x	0	0,04	0,08	0,12	0,16	0,2	0,24	0,28	0,32	0,36	0,4	0,44	0
$(T_1+T_2)/2$	1,794280586	1,771564487	1,748242305	1,724262775	1,699566965	1,674086578	1,647741715	1,620437921	1,592062176	1,562477342	1,531514278	1,498960297	1,794280586
h	0,234314575	0,234314575	0,234314575	0,234314575	0,234314575	0,234314575	0,234314575	0,234314575	0,234314575	0,234314575	0,234314575	0,234314575	0,234314575
alpha [rad]	0,785398163	0,806967931	0,830423629	0,856058085	0,884232265	0,915397217	0,950125359	0,989156267	1,03346568	1,084373219	1,14371774	1,214157882	0,785398163
alpha [°]	45	46,23585667	47,57976916	49,04851527	50,66277689	52,44839709	54,43817308	56,67447937	59,21322173	62,13000886	65,53019948	69,56612228	45
	2,35619449	2,334624722	2,311169024	2,285534569	2,257360389	2,226195437	2,191467295	2,152436387	2,108126974	2,057219435	1,997874913	1,927434772	2,35619449
	0,451026812	135	133,7641433	132,4202308	130,9514847	129,3372231	127,5516029	125,5618269	123,3255206	120,7867783	117,8699911	114,4698005	110,4338777
	2,690565842	0,785398163	0,806967931	0,830423629									135
	3,141592654	45	46,23585667	47,57976916	49,04851527	50,66277689	52,44839709	54,43817308	56,67447937	59,21322173	62,13000886	65,53019948	69,56612228

Mithilfe von PythonTeX ist es möglich, dies in einem Dokument zu realisieren.

```
\begin{pycode}
# Initialisierung
l,g,phi = 0.8,9.81,np.pi/4
# Tabelle wie Excel
makeTableFromFunctionq(0,.05,lambdax : np.pi *
(np.sqrt(l/g) + np.sqrt(l*(1-x)/g)),
mi=6,mu='\\text{m}',fu='\\text{s}',rnd=2)
\end{pycode}
```


4. Online-Einsatz während und nach Corona

Die Lernplattform Moodle bietet mit der Aktivität Test die Möglichkeit, ein interaktives Aufgabenblatt mit verschiedenen Aufgabentypen zu realisieren.

ab hier nicht diese Aktivität



sondern diese



Speziell mit dem PlugIn STACK, welches das CAS Maxima verwendet, ist es möglich, mathematische Ausdrücke algebraisch zu vergleichen. Zur Implementierung ist es also nötig, die Maxima-Syntax zu verwenden. Hierzu empfiehlt es sich, Maxima als Standalone-Desktop-Anwendung zu installieren. So kann man komplexere Ausdrücke vorab testen.

Eine STACK-Aufgabe erscheint den Studierenden in folgender Form: Die Zahlenwerte in den Rechtecken sind Zufallsparameter, deren Wertebereich in der Aufgabe vordefiniert ist.

Vorschau Frage A003_Zusammentreffen - Google Chrome
 lemen.h-da.de/question/preview.php?id=4644177&cmid=529315&behaviour=deferredfeedback&maxmark=3&cor...
 Torsten-Karl Stempel
 h_da Physik EIT Stempel SS2021
 Home Meine Kurse Dieser Kurs Kursbereiche Infos Navigation
 Frage 1
 Bisher nicht beantwortet.
 Erreichbare Punkte: 3,00
 Von zwei **34.0 [km]** weit auseinander liegenden Ortschaften gehen zwei Personen einander entgegen, die eine **3.3 [km/h]** e...
 t **6.3 [km/h]** e...
 • Nach wie vielen Stunden treffen sie zusammen?
 [h]
 • Welchen Weg legt Person 1 zurück?
 [m]
 • Welchen Weg legt Person 2 zurück?
 [m]
 Bitte benutzen Sie bei den Antworten nur die Dezimalzahlen gerundet auf 2 Nachkommastellen.
 Bitte keine Zeichen, keine Brüche, d.h. statt 1/3 schreiben Sie z.B. 0.33 (mit Dezimaltrennpunkt)
 Denken Sie daran das Ergebnis in SI-Einheiten anzugeben und mit korrektem Vorzeichen anzugeben.
 Die Einheit selbst muss nicht angegeben werden.
 Nochmal beginnen Sichern Richtige Lösung Absenden und beenden
 Technische Information

Die Studierenden geben die Ergebnisse hier als gerundete (Physik!) Zahlenwerte ein. Die Aufgabe ist so konfiguriert, dass für jedes Aufgabenfeld separat Punkte

vergeben werden. Man erhält auch Punkte für richtige Ergebnisse, wenn nicht alle Felder ausgefüllt oder Fehler gemacht wurden.

Die Aufgabe ist in ein Aufgabenblatt (s.o. Aktivität Test) eingebettet, das Studierende beliebig aufrufen und bearbeiten können, bis die Bearbeitungsfrist abgelaufen ist. Am Ende werden alle erfassten Ergebnisse ausgewertet; die Studierenden müssen nicht extra sichern.

5. Kombination mit `moodle.sty`

Aufgabenblätter im PDF-Format können ausgedruckt, in Präsenzveranstaltungen verteilt und bearbeitet werden. Die Web-Darstellung in Moodle kann man nur unzureichend als PDF-Aufgabenblatt abspeichern oder ausdrucken. Das LaTeX-Paket `moodle.sty` stellt eine Möglichkeit zur Kombination dar. Die LaTeX-Syntax wird um Befehle zur Erstellung von verschiedenen Moodle-Fragetypen erweitert und bei der Erzeugung der Ausgabe wird eine XML-Datei erzeugt, die man in Moodle importieren kann.

```

\documentclass[12pt,handout]{article}
\usepackage{moodle} % Paket moodle
%\usepackage{tikz}
%\usepackage{graphicx}
\begin{document}
\begin{quiz}{Komplexe Zahlen} % Frageteil
  \begin{multi}{Frage 1 Wurzel aus Quadrat}
    Was ist  $\sqrt{(-1)^2}$ ?
    \item[fraction=100, feedback={
      Durch das Quadrat wird die (-1) positiv.
      Die Wurzel einer positiven Zahl ist auf
      jeden Fall positiv.}]* 1
    \item -1
    \item Man kann das nicht berechnen,
      weil es keine Wurzel aus negativen
      Zahlen gibt
  \end{multi}
\end{quiz}
\end{document}

```

Leider ist es nicht möglich, auch STACK-Aufgaben damit zu erzeugen. Allerdings kann man **moodle.sty** wieder mit PythonTeX kombinieren und so Aufgaben parametrisiert erzeugen. Diese werden dann als einzelne Aufgaben mit festen Werten in die Moodle-Aufgabendatenbank importiert und man erstellt ein Aufgabenblatt, das Aufgaben aus einem Themenbereich zufällig auswählt.

6. Fazit

Parametrisierte Aufgaben mit Moodle und STACK stellen ein mächtiges Werkzeug dar und waren in der Online-Phase eine große Hilfe. Sie können allerdings nicht alle Aufgabenbereiche abdecken und die klassischen Übungen, in denen gemeinsam gerechnet wird, nicht ersetzen. Sie ergänzen aber ideal die reguläre Präsenzlehre, da die Studierenden zusätzliche Übungsmöglichkeiten erhalten.

Vorteile sind aus meiner Sicht:

- STACK ist Dank CAS in der Lage algebraische Äquivalenz zu erkennen.
- Mit **moodle.sty** kann man bestehende Aufgabenblätter einfach in Moodle-Aufgaben überführen.
- Moodle präsentiert Dank LaTeX mathematische Ausdrücke gut lesbar.
- Studierende können zeitlich und räumlich unabhängig auf zusätzliche Übungen zugreifen.
- Lehrende können gemeinsam Aufgabendatenbanken pflegen und nutzen.

Nachteile sind:

- Die Aufgabenerstellung ist aufwändig und STACK erfordert eine gewisse Einarbeitung.
- Die Fehlersuche kann teilweise mühsam sein.
- Die Auswertung ist nicht allzu komfortabel (z.B. muss man beim Excel-Download Text- in Zahlenfelder umwandeln)
- Die Studierenden müssen bei der Eingabe einige Syntaxregeln berücksichtigen.
- STACK-Aufgaben werden nicht in der Moodle-App angezeigt.

Links

1. <https://moodle.de/>, <https://docs.moodle.org/310/de/Hauptseite>
2. <https://ctan.org/pkg/pythontex>
3. https://docs.moodle.org/310/de/Fragetyp_STACK
4. <https://docs.stack-assessment.org/content/2019-STACK-Guide.pdf>.
5. <https://maxima.sourceforge.io/de/index.html>
6. <https://ctan.org/pkg/moodle>

Autor

Prof. Dr. Torsten-Karl Stempel

fbmn Fachbereich Mathematik und Naturwissenschaften
h_da Hochschule Darmstadt
Schöfferstr.3
64295 Darmstadt

E-Mail: torsten-karl.stempel@h-da.de

Georg Hoever

Integration von Mumie-Aufgaben in der höheren Mathematik-Ausbildung

Zusammenfassung: Die Plattform Mumie bietet (ähnlich wie STACK) die Möglichkeit, randomisierte Aufgaben zur Verfügung zu stellen, die automatisch korrigiert werden. An der FH Aachen werden entsprechende Aufgaben als Pflicht- und Bonus-Aufgaben im Rahmen der höheren Mathematik für Elektrotechnik und (Wirtschafts-)Informatik eingesetzt.

Im Artikel werden die Plattform und Einsatzmöglichkeiten vorgestellt und von den Erfahrungen aus dem Einsatz berichtet.

1. Einführung

Am Fachbereich Elektro- und Informationstechnik der FH Aachen beginnen jährlich ca. 400 Informatik-, 100 Elektrotechnik- und 70 Wirtschaftsinformatik-Studierende ihr Studium. Im ersten Semester gibt es ein mit 9 Leistungspunkten und 8 Semesterwochenstunden recht großes gemeinsames Mathematik-Modul. Im zweiten Semester ist die Mathematik getrennt einerseits für die Informatik und Wirtschaftsinformatik (6 Leistungspunkte, 5 Semesterwochenstunden), andererseits für die Elektrotechnik (5 Leistungspunkte, 4 Semesterwochenstunden).

Eine Motivation für die Studierenden, sich regelmäßig mit dem dargebotenen Stoff zu beschäftigen wird an vielen Universitäten durch wöchentliche Abgabenaufgaben realisiert, die von studentischen Hilfskräften aus den Mathematik-Studiengängen korrigiert werden. Da es an der FH Aachen – wie an vielen Fachhochschulen – keine originären Mathematik-Studiengänge gibt, fehlt ein Zugriff auf entsprechend versierte studentische Hilfskräfte. Unter den (Wirtschafts-)Informatik- und Elektrotechnik-Studierenden gibt es zwar natürlich auch solche, die sehr gut in Mathematik sind, aber die Erfahrung zeigt, dass für eine Korrektur – insbesondere wenn es verschiedene Lösungswege oder -darstellungen gibt – eine solide Mathematik-Ausbildung sinnvoll ist. Besonders ärgerlich ist, wenn sehr gute Studierende kreative Lösungswege gehen, die von einem nicht so versierten Korrektor fälschlicherweise angestrichen werden.

Abhilfe schaffen automatisch korrigierte Aufgaben, bei denen ein hinter der Korrektur befindliches Computeralgebra-System auch verschiedene Lösungsdarstellungen als richtig bewerten kann. Randomisierte Aufgaben ermöglichen dann für jede(n) Studierende(n) individuelle Aufgaben. Beweis-Aufgaben sind auf diese Weise zwar nicht unmittelbar machbar, aber das ist bei der Mathematik für Ingenieure an Fachhochschulen auch nicht der hauptsächliche Fokus; die Mathematik ist dort hauptsächlich Werkzeug und daher sind Rechen-Aufgaben ein guter Weg, den gelernten Stoff umzusetzen und zu vertiefen.

Am Fachbereich Elektro- und Informationstechnik der FH Aachen wird die „Mumie“-Plattform genutzt, um entsprechende Aufgaben zu entwickeln und bereit zu stellen. Im nächsten Abschnitt wird diese Plattform genauer vorgestellt. In Abschnitt 3 wird die konkrete Umsetzung beschrieben.

2. Die Mumie-Plattform

Die „Mumie“ ist eine LMS-unabhängige Plattform für e-Learning insbesondere im Bereich Mathematik und Informatik. Sie ist beispielsweise auch Basis für die weit verbreiteten online-Kurse OMB+ und HM4MINT (s. [1], [2]). Es gibt eine umfangreiche qualitätsgesicherte Aufgabensammlung sowie Plugins zur Integration in Moodle und Ilias. Entwickelt und betreut wird die Plattform durch die Firma Integral-Learning ([3]), einem Spinoff der TU Berlin, gegründet von Prof. Dr. Ruedi Seiler.

Die Plattform bietet (ähnlich wie STACK) parametrisierbare Aufgaben und nutzt ein dahinter liegendes Computeralgebra-System für Berechnungen und Kontrolle der Nutzereingaben. Mit JSXGraph können interaktive Grafiken genutzt werden. Die Programmierung von Aufgaben ist im Autoren-Tool durch eine Latex-basierte Syntax möglich. Beispielsweise erzeugt der Quelltext in Abbildung 1 die im Kasten dargestellte Aufgabe.

```

\begin{question}
  \type{input.number}
  \field{rational}
  \lang{de}{\text{
    Betrachtet wird die Folge  $(a_n)_{n \in \mathbb{N}}$  mit  $a_n = \text{\var{folge}}$ . \\
    1. Geben Sie eine Vermutung für den Grenzwert  $a$  der Folge an:  $a = \text{\ansref}$ . \\
    2. Bestimmen Sie das minimale  $N \in \mathbb{N}$ , so dass  $|a_n - a| < \text{\var{eps}}$  für
    alle  $n \geq N$  ist:  $N = \text{\ansref}$ .
  }}
  \begin{variables}
    \randint[Z]{a}{-8}{8}
    \randint[Z]{b}{-8}{8}
    \randadjustIf{b}{abs(a)=abs(b)}
    \function[normalize]{folge}{(a*n+b)/n}
    \randint{c}{2}{8}
    \function[normalize]{eps}{1/(c*100)}
    \function{N}{c*100*abs(b)+1}
  \end{variables}

  \begin{answer}
    \solution{a}
  \end{answer}
  \begin{answer}
    \solution{N}
  \end{answer}
\end{question}

```

Betrachtet wird die Folge $(a_n)_{n \in \mathbb{N}}$ mit $a_n = \frac{3n+5}{n}$.

1. Geben Sie eine Vermutung für den Grenzwert a der Folge an: $a =$.

2. Bestimmen Sie das minimale $N \in \mathbb{N}$, so dass $|a_n - a| < \frac{1}{800}$ für alle $n \geq N$ ist: $N =$.

Abbildung 1: Quellcode und erzeugte Aufgabe

Weitere beispielhafte Gestaltungselemente sind Mengen (s. Abbildung 2 links) und Matrizen (s. Abbildung 2 rechts).

(a) Welche Nullstellen hat die Funktion $\sin(x)$ im Intervall $[-2\pi; 3\pi[$?
 Hinweis: Sie können die Werte als Vielfache von π angeben, z.B. "3/2pi".

Die gesuchte Nullstellenmenge ist $\{ -6,28 ; [-\pi] \}$

(b) Für welche x -Werte im Intervall $[-2\pi; 4\pi[$ hat die Funktion $\sin(x)$ ihren maximalen Wert?
 Hinweis: Sie können die Werte als Vielfache von π angeben, z.B. "3/2pi".

Die gesuchte Menge der x -Werte ist ?

(b) Bestimmen Sie zu $A = \begin{pmatrix} 1 & -2 \\ -4 & 0 \end{pmatrix}$ und $B = \begin{pmatrix} -5 & -1 \\ -1 & 3 \end{pmatrix}$ die beiden Matrix-Produkte $A \cdot B$ und $B \cdot A$:

$A \cdot B = \begin{pmatrix} ? & ? \\ ? & ? \end{pmatrix}$

$B \cdot A = \begin{pmatrix} ? & ? \\ ? & ? \end{pmatrix}$

(c) Bestimmen Sie das Format und die Einträge zur folgenden Matrixmultiplikation:

$\begin{pmatrix} -2 & 0 \\ 1 & 1 \\ 0 & -3 \end{pmatrix} \cdot \begin{pmatrix} 2 & 1 & 2 & 3 \\ 4 & 2 & -4 & 1 \end{pmatrix} = ?$

Format: x

Abbildung 2: Mengen und Matrizen als Gestaltungselemente

3. Konkrete Umsetzung

Um die Studierenden zu motivieren, sich regelmäßig mit dem dargebotenen Stoff zu beschäftigen, gibt es wöchentliche elektronische Abgabeaufgaben. Gemäß der Devise „Zuckerbrot und Peitsche“ gibt es dabei zum einen Bonus-Aufgaben, mit denen Bonuspunkte für die das Modul abschließende Klausur erworben werden können, und zum anderen Pflicht-Aufgaben, deren erfolgreiche Bearbeitung Voraussetzung ist, um für die abschließende Klausur zugelassen zu sein.

Bei den Pflicht-Aufgaben hat es sich bewährt, wöchentlich Aufgaben mit in Summe 40 Punkten zu stellen, von denen mindestens 30 Punkte erreicht werden müssen; außerdem haben die Studierenden zwei „Joker“: Eine bis zu zweimalige nicht ausreichende Bearbeitung der Pflicht-Aufgaben während des Semesters hat keine negativen Konsequenzen. Diese Joker sind beispielsweise für Krankheitsfälle gedacht; damit erspart man sich viel Arbeit mit Attesten. Die Pflicht-Aufgaben sind so konzipiert, dass sie den größten Teil des dargebotenen Stoffs abdecken. Beispielsweise gibt es zum Themenkomplex „Folgen“ in einer Woche die folgenden 6 Aufgaben:

1. Folgenreihe aus einer grafisch dargestellten Folge ermitteln (s. Abbildung 4 links).
2. Multiple-Choice-Aufgabe zur Monotonie und Beschränktheit (s. Abbildung 4 rechts).
3. Berechnung, ab welchem Folgenindex eine gegebene Folge in einer bestimmten ε -Umgebung des Grenzwertes ist (s. Abbildung 5 links).
4. Berechnung einfacher Grenzwerte (s. Abbildung 5 rechts).
5. Berechnung von Grenzwerten bei rekursiv definierten Folgen.
6. Berechnung komplizierterer Grenzwerte, z.B. bei

$$\sqrt{6n^2 - 5n} - \sqrt{6n^2 - n + 3} \quad \text{oder} \quad \left(1 - \frac{4}{n}\right)^n.$$

Die Bonus-Aufgaben sind meist Kontext-bezogene Aufgaben, die eine komplexere oder mehrstufige Berechnung erfordern, s. Abbildung 6.

(a) Die in der folgenden Grafik dargestellten Folgen haben die Form $a_n = c + d \cdot \frac{1}{n}$ bzw. $a_n = c + d \cdot \frac{(-1)^n}{n}$ mit $c, d \in \mathbb{Z}$. Geben Sie die konkrete Vorschrift an.

Die rot markierte Folge wird beschrieben durch $a_n = \boxed{?}$.

Die blau markierte Folge wird beschrieben durch $a_n = \boxed{?}$.

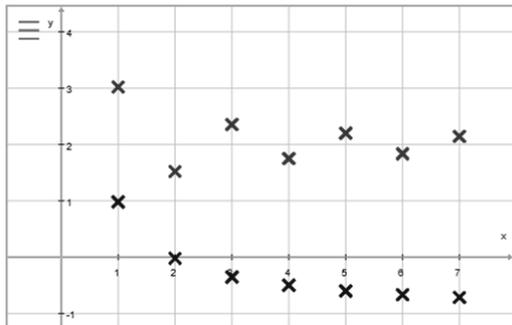


Abbildung 4: Pflicht-Aufgaben zu Folgen

(a) Betrachtet wird die Folge $(a_n)_{n \in \mathbb{N}}$ mit $a_n = \frac{7n+2}{n}$.

1. Geben Sie eine Vermutung für den Grenzwert a der Folge an:

$a = \boxed{7}$.

2. Bestimmen Sie das minimale $N_1 \in \mathbb{N}$, so dass $|a_n - a| < \frac{1}{10}$ für alle $n \geq N_1$ ist:

$N_1 = \boxed{?}$.

3. Bestimmen Sie das minimale $N_2 \in \mathbb{N}$, so dass $|a_n - a| < \frac{1}{600}$ für alle $n \geq N_2$ ist:

$N_2 = \boxed{?}$.

Kreuzen Sie alle wahren Aussagen an.



<input type="checkbox"/>	Die rekursiv definierte Folge $(a_n)_{n \in \mathbb{N}}$ mit $a_1 = 2$ und $a_{n+1} = -2a_n - 5$ ist monoton.
<input type="checkbox"/>	Die Folge $(a_n)_{n \in \mathbb{N}}$ mit $a_n = 5n + 5$ ist monoton wachsend.
<input type="checkbox"/>	Die Folge $(a_n)_{n \in \mathbb{N}}$ mit $a_n = (-1)^n \cdot n$ ist nach oben beschränkt.
<input type="checkbox"/>	Die Folge $(a_n)_{n \in \mathbb{N}}$ mit $a_n = \frac{(-1)^n}{n}$ ist beschränkt.
<input type="checkbox"/>	Jede monoton wachsende Folge ist nach unten beschränkt.
<input type="checkbox"/>	Die rekursiv definierte Folge $(a_n)_{n \in \mathbb{N}}$ mit $a_1 = 2$ und $a_{n+1} = 4a_n + 3$ ist nicht monoton.

(a) Bestimmen Sie die angegebenen Grenzwerte in $\mathbb{R} \cup \{\pm\infty\}$.

Hinweis: Geben Sie alle Werte vollständig gekürzt ein. Für die Eingabe von ∞ verwenden Sie bitte "infy" und für die Eingabe von $-\infty$ "-infy".

1. $\lim_{n \rightarrow \infty} \frac{-2n^2 - 3n - 5}{-5n^2 - 5n + 4} = \boxed{?}$

2. $\lim_{n \rightarrow \infty} \frac{-5n^2 + 3}{n^2 + 4} = \boxed{?}$

3. $\lim_{n \rightarrow \infty} \frac{-6n^4 + 5}{5n^2 - 3} = \boxed{?}$

4. $\lim_{n \rightarrow \infty} \frac{2n^8 + 5}{-(n^2 - 1)^4} = \boxed{?}$

Abbildung 5: weitere Pflicht-Aufgaben zu Folgen

Ein tiefer Brunnen soll gebaut werden. Um die Kosten für die Grabung des Brunnens besser abschätzen zu können, wird die Bohrung in jeweils 10 Meter tiefe Abschnitte eingeteilt. Die Bohrung und Ausgrabung des ersten Abschnittes (0 bis 10 Meter) wird 400 Euro kosten. Jeder weitere 10 Meter große Abschnitt wird jeweils 7 Prozent mehr kosten als der direkt vorherige.



Hinweis: Runden Sie in den Aufgabenteilen b) - d) bei der Eingabe jeweils sinnvoll auf ganze Zahlen.

(a) Obiger Sachverhalt lässt sich anhand einer Folge beschreiben: a_n sollen die Kosten für den n -ten Bohrabschnitt angeben.

1. Geben Sie eine rekursive Vorschrift an: $a_1 = \boxed{?}$, $a_{n+1} = \boxed{?} \cdot a_n$

2. Geben Sie eine Vorschrift der Folge an, um die Folgenglieder direkt berechnen zu können: $a_n = \boxed{?}$.

(b) Berechnen Sie, wie teuer die Bohrung des 20. Abschnittes wird.

Die 20. Bohrung wird rund $\boxed{?}$ Euro kosten.

(c) Entscheiden Sie, wie tief der Brunnen werden kann, wenn der letzte Bohrabschnitt maximal 1000 Euro kosten darf.

Der Brunnen darf $\boxed{?}$ Meter tief werden.

(d) Ermitteln Sie, wie teuer der erste Bohrabschnitt maximal sein kann, damit der 29. Bohrabschnitt nicht über 1000 Euro kostet. Gehen Sie weiterhin davon aus, dass der jeweils nächste Bohrabschnitt 7 Prozent teurer wird. Runden Sie ihr Ergebnis auf ganze Euro.

Der erste Bohrabschnitt darf maximal $\boxed{?}$ Euro kosten.



Abbildung 6: Bonus-Aufgabe zu Folgen

Für die Bearbeitung der Pflicht- und Bonus-Aufgaben haben die Studierenden eine Woche Zeit. Hilfestellungen erhalten sie durch Tipps in den Vorlesungen und Übungen und durch Tutorien, in denen Studierende höheren Semesters individuell Hilfestellungen geben. Es hat sich bewährt, dass die Studierenden beliebig viele Eingabe-Versuche (mit Korrektur) haben, d.h., nach Eingabe einer falschen Lösung erhalten sie das Feedback, dass die Antwort nicht richtig ist, und können eine neue Eingabe testen. Die Erfahrung zeigt, dass die Gefahr, dass die Studierenden ohne viel nachzudenken einfach Lösungen probieren, gering ist; vielmehr überwiegt der positive Effekt, dass die Studierenden Dinge ausprobieren und bei Rückmeldung einer fehlerhaften Eingabe selbst ihre Denk- oder Rechenfehler suchen.

Nach Abgabeende bleiben die Aufgaben weiterhin sichtbar, können aber zunächst nicht weiter editiert werden. Zur Klausurvorbereitung werden die Aufgaben dann wieder freigeschaltet; dabei gibt es einen zusätzlichen Button „Neue Daten“, mit denen die Studierenden sich neue randomisierte Werte für die einzelnen Aufgaben erzeugen können, wenn sie die Rechnung mit anderen Werten üben möchten.

Die Modul-Evaluation zeigt, dass das Ziel, mit den elektronischen Aufgaben die regelmäßige Beschäftigung zu fördern, erreicht wird: „Der e-Test hat einen mehr oder weniger gezwungen sich jede Woche mit den aktuellen Themen auseinander zu setzen und nicht erst kurz vor der Klausur alles im Hau-Ruck-Verfahren durchzuarbeiten, was einem dabei hilft, am Ball zu bleiben, und dann die nötigen Grundlagen für die weiteren Vorlesungen zu beherrschen.“

Literaturverzeichnis

1. www.ombplus.de
2. www.hm4mint.nrw
3. www.integral-learning.de

Autor

Prof. Dr. rer. nat. Dr.-Ing. Georg Hoever
Fachbereich Elektro- und Informationstechnik
FH Aachen
Eupener Str. 70
D-52066 Aachen
E-Mail: hoever@fh-aachen.de

Patricia Graf, Heiko Knospe, Andreas Schwenk

Mathe:buddy - eine Gamifizierte Lern-App für die Höhere Mathematik

Zusammenfassung. Mathe:buddy ist eine gamifizierte Lern-App für Studierende des ersten und zweiten Semesters Mathematik in MINT-Studiengängen. Sie kann als Begleitung von Vorlesung und Übung verwendet werden, erleichtert den Einstieg und die Wiederholung von Themen und ermöglicht eine eigenständige Kontrolle des Lernstands. Die Lern-App stellt kleinere Lerninhalte in den Vordergrund, die durch eine Baumstruktur verknüpft sind. Die Motivation der Studierenden soll durch spielerische Elemente gesteigert werden. In diesem Beitrag berichten wir über den aktuellen inhaltlichen und technischen Entwicklungsstand.

Einleitung

Für viele Studienanfänger:innen in den MINT-Studiengängen stellt die Höhere Mathematik eine Herausforderung dar. Die Vorkenntnisse aus der Schule und die Studienanforderungen der Hochschule weichen häufig voneinander ab [1]. Vorkurse und (Online-)Brückenkurse (z.B. [2]) werden nur teilweise genutzt. Heterogene Vorkenntnisse können zudem die anfängliche Synchronisation erschweren. Die mathematischen Themen werden von der Zielgruppe oft als abstrakt wahrgenommen und die Bedeutung der zu erwerbenden Kompetenzen für andere Studienfächer wird nicht immer deutlich. Den Studierenden fehlen oft Anreize und Möglichkeiten, sich regelmäßig und aktiv einzubringen und die eigenen Kenntnisse zu überprüfen. Eine individuelle Begleitung und Förderung von Studierenden seitens der Hochschule ist häufig nur eingeschränkt möglich. Für viele Studierende ist es zudem herausfordernd, die Lernmotivation über einen längeren Zeitraum aufrechtzuerhalten. Das hier vorgestellte Projekt “mathe:buddy” adressiert die praktische Umsetzung durch eine gamifizierte Lern-App, die Studierende motiviert, begleitet und individuell fördert. Durch interaktives Training und Gamifizierung kann der Lernprozess unterstützt und unterschiedliche Lerntypen erreicht werden. In anderen Kontexten werden gamifizierte Apps bereits erfolgreich eingesetzt [4, 5]. In den folgenden Kapiteln wird das inhaltliche Konzept und die technische Realisierung erläutert.

Inhaltliches Konzept

Mathematische Themen werden in kleine Lerninhalte (sogenannte “Level”) aufgeteilt. Ein Level hat jeweils einen dreiteiligen Aufbau (siehe Abbildung 1). Jedes Level beginnt mit einer einführenden Aufgabe; diese Aufgabe ist ohne das inhaltliche Wissen des Levels lösbar. Sie gibt den Studierenden einen intuitiven Ansatz sich selbst neue Inhalte zu erschließen. Der formale Lerninhalt stellt den zweiten Level-Abschnitt dar, beispielsweise eine mathematische Definition oder ein mathematischer Satz. Diese werden kurz

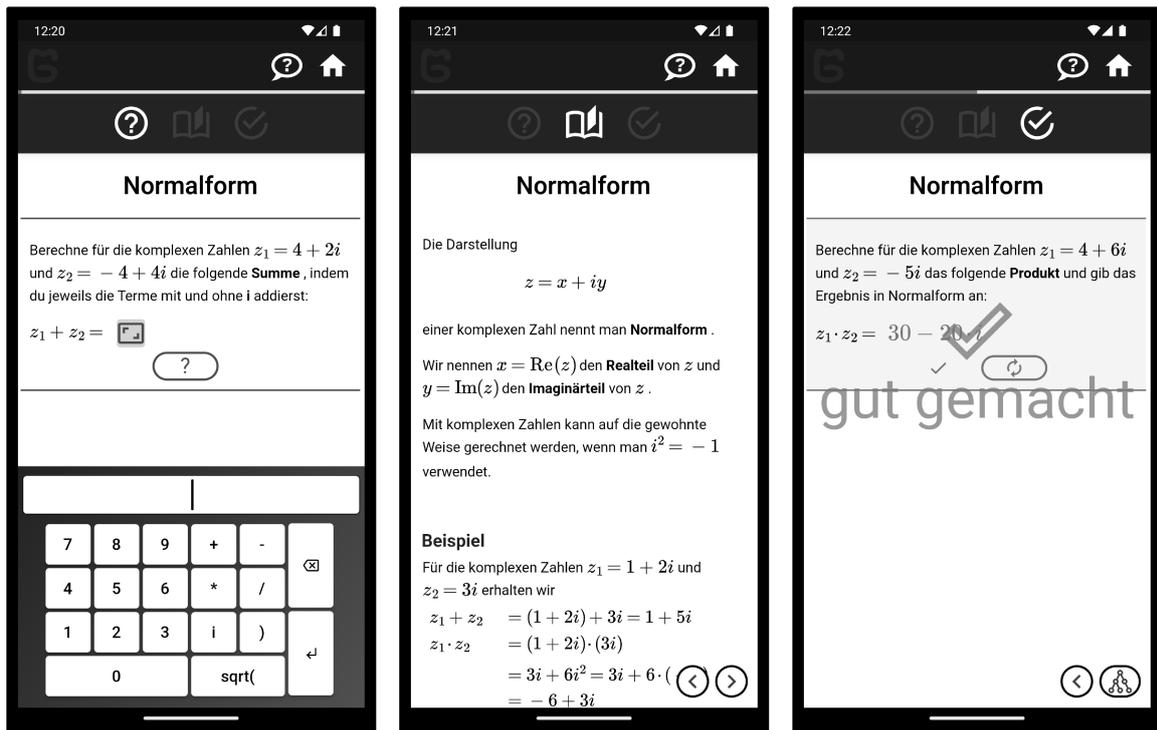


Abb. 1: Einführungsaufgabe, Vorlesungstext mit Beispiel, Abschlussaufgabe

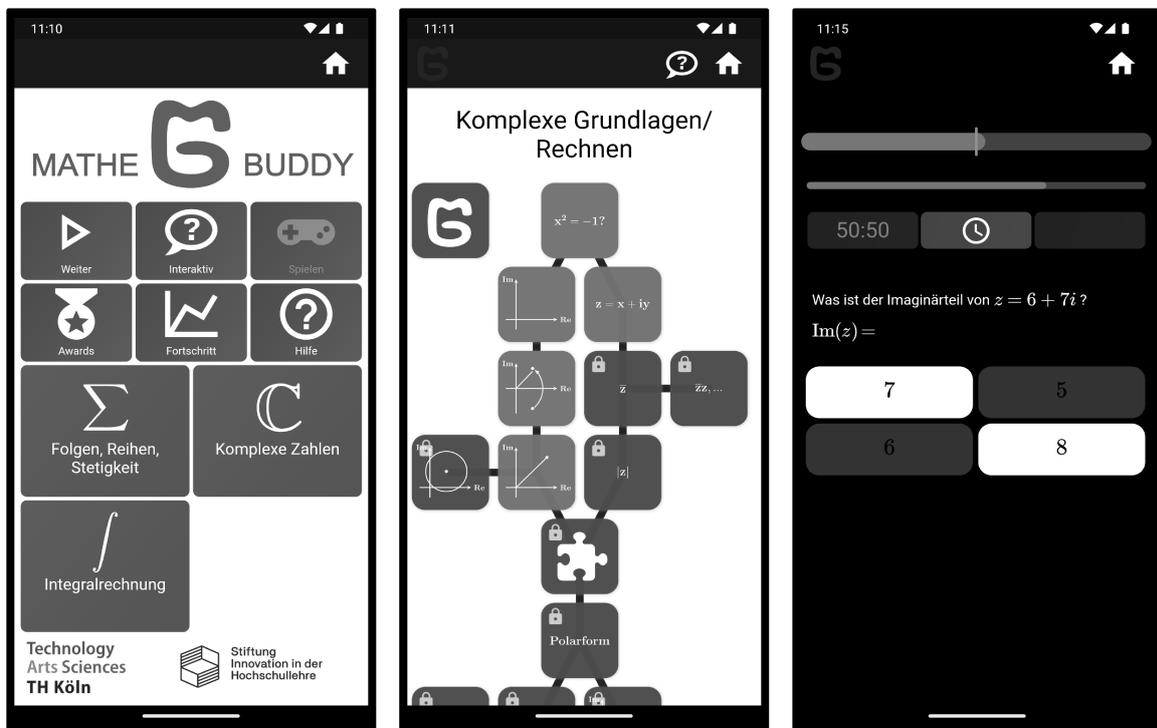


Abb. 2: Startseite, Levelauswahl, Event

und verständlich eingeführt und mit einem Beispiel abgeschlossen. Im letzten Teil eines Levels wird auf diesen Lerninhalt durch eine vertiefende Aufgabe eingegangen.

Die einzelnen Level eines Themas werden durch eine Baumstruktur miteinander verknüpft (siehe Abbildung 2 Levelauswahl). Die Studierenden starten ein Thema mit dem obersten Level und bewegen sich entlang des Baums nach unten. Ein Level wird freigeschaltet, wenn alle vorherigen verknüpften Level erfolgreich gespielt wurden. Es können außerdem Level aus anderen Kapiteln vorausgesetzt werden. Zum Beispiel benötigt das Level “Komplexe Folgen” als Voraussetzung unter anderem Level aus dem Gebiet der “Reellen Folgen”. Diese Struktur hilft den Studierenden Themen einzuordnen und Zusammenhänge besser zu erfassen. Ein Level kann drei verschiedene Zustände annehmen. Ist ein Level rot, so wurde es noch nicht gespielt. Das Schloss gibt dabei an, ob es bereits erfolgreich freigeschaltet wurde. Gelbe Level wurden bereits gespielt, aber die Aufgaben noch nicht oder nur teilweise richtig gelöst. Wenn beide Aufgaben korrekt gelöst wurden, erscheint das Level in grün.

Neben den inhaltlichen Level eines Spielbaums existieren außerdem “Events” (siehe Abbildung 2 Event). Diese werden optisch durch ein Puzzleteil dargestellt. Ein Event ist ein zeitbasiertes Spiel mit Aufgaben aus dem vorherigen Teilthema. Durch dieses Konzept werden Themen spielerisch wiederholt und levelübergreifende Inhalte abgefragt. Die Aufgaben sind dabei meistens Multiple-Choice Aufgaben, bei denen jeweils nacheinander eine Zeit abläuft. Die Bewertung erfolgt nach dem “high-risk-high-reward” Prinzip. Schnelle richtige Antworten werden dementsprechend mit vielen Punkten belohnt und für schnelle falsche Antworten werden mehr Punkte abgezogen. Das Ziel des Spieles ist es über einer gewissen Grenze an Punkten zu bleiben. Diese Grenze existiert in drei Schwierigkeitsstufen, für jede erfolgreiche Stufe erhält man einen Stern. Ein weiteres spielerisches Element der Events bilden die drei verschiedenen Joker: a) 50/50-Joker: Löscht die Hälfte der möglichen Antworten in Multiple-Choice Aufgaben. b) Punktemultiplikator-Joker: Verdoppelt die Punkte in einer Aufgabe. c) Zeitmultiplikator-Joker: Verdoppelt die Zeit in einer Aufgabe. Die Joker erhält man durch erfolgreiches Abschließen eines inhaltlichen Levels. Um die Motivation zu steigern, erhält man anschließend mit einer gewissen Wahrscheinlichkeit Joker bei Wiederholung von randomisierten vertiefenden Aufgaben eines Levels.

Technische Realisierung

Die technische Umsetzung gliedert sich in zwei Bereiche.

1. Entwicklung der App für die Höhere Mathematik. Die Verwendung einer einheitlichen Codebasis für die nativen iOS und Android Apps und einer Webversion wird durch das Framework “Flutter” (Google) ermöglicht (vgl. [3]).
2. Entwicklung eines webbasierten Workflows als Entwicklungsumgebung für Dozierende, die neue Kursteile erstellen. Technisch ist dies nicht auf Inhalte der Höheren Mathematik beschränkt.

Der Entwurf der Softwareumgebung zielt auf eine nachhaltige Nutzung und Weiterentwicklung ab. Die Implementierung erfolgt als Open Source und ist unter <https://github.com/mathebuddy> verfügbar. Abbildung 3 zeigt die Softwarekomponenten.

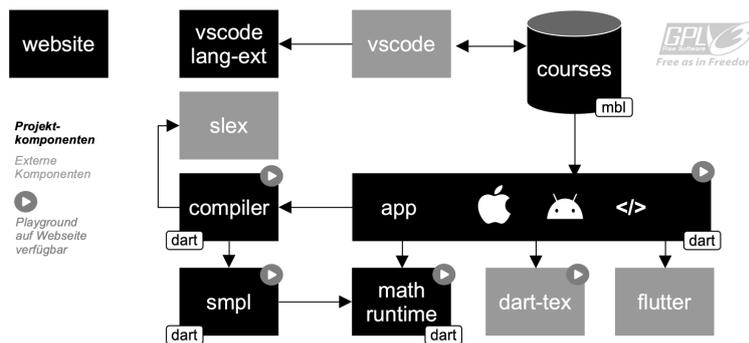


Abb. 3: Komponentendiagramm

Die Erstellung von Kursteilen erfolgt mittels der im Rahmen des Projekts entwickelten Definitionssprache MBL (Mathe:Buddy Language). Aus den von den Lehrenden beschriebenen Inhalten wird automatisiert eine Wissensdatenbank aufgebaut, die beispielsweise für den integrierten Chatbot genutzt wird. MBL ist quelloffen und kann prinzipiell auch als textbasiertes, formales Austauschformat für andere Onlinekurse eingesetzt werden. Der Fokus der Sprachentwicklung liegt auf Einfachheit und Erlernbarkeit, sodass Dozierende sich auf die didaktischen Aspekte der entwickelten Kurse fokussieren können. Auf einen syntaktischen Overhead in Form von Sonderzeichen wird weitgehend verzichtet. Die beiden Abbildungen 4 und 5 zeigen Teile eines Levels aus Entwicklersicht, sowie dessen Debug-Vorschau im Simulator.

MBL ist hierarchisch strukturiert und setzt (ähnlich wie die Programmiersprache Python) auf Einrückungen, um Abhängigkeiten auch auf Quelltextebene anschaulich auszudrücken. Die Sprache folgt weitgehend dem deskriptiven Paradigma. Das Layout von Paragraphen ist an *Markdown* angelehnt. Der mathematische Formelsatz stützt sich auf $\text{T}_{\text{E}}\text{X}$. Die Schlüsselwörter entsprechen den Anforderungen der Domäne. Beispielsweise wird eine Übungsaufgabe per EXERCISE eingeleitet, CENTER zentriert die nachfolgenden Textbausteine, DEFINITION beginnt eine mathematische Definition, usw. Für die Konfiguration des automatischen Lernmanagements steht eine Zahl optional einsetzbarer Schlüsselwörter bereit. Beispielsweise kann markiert werden, ob eine Aufgabe verstärkt bei der Wiedervorlage im Rahmen des automatischen Lernmanagements eingesetzt werden soll. Die in den Kursen integrierten Übungsaufgaben können durch ihre Randomisierung meist wiederholt gespielt werden. Hierzu wird in MBL die eigens für das Projekt entwickelte formale Sprache SMPL (Simple Math Programming Language) eingebettet. Diese dient neben der Erzeugung von Zufallswerten auch der Berechnung der Musterlösung. Die Mathematik-Engine von SMPL unterstützt zum Beispiel das automatische Differenzieren; zum Teil steht auch das symbolische Rechnen mit Termoptimierung zur Verfügung. Auf eine Vielzahl integrierter Methoden kann zurückgegriffen werden. Insbesondere die Generierung von Zufallsvariablen ist sehr ausdrucksstark formulierbar. Beispielsweise erzeugt der Ausdruck $A/B/C = \text{randZ}\langle 3,4\rangle(-5,5)$ drei randomisierte 3×4 -Matrizen, die wertmäßig verschieden sind.

Die webbasierte Entwicklungsumgebung, Hilfeseiten sowie weitere Playgrounds sind unter <https://mathebuddy.github.io> öffentlich nutzbar.

```

u1-010-intro.mbl M X u1-020-gausscheZahlenebene.mbl u1-03
hm1 > komplex > u1-010-intro.mbl
18 DEFINITION @def:intro
19 Über den reellen Zahlen kannst du Gleichungen wie  $x^2=-1$  nicht
    lösen, da keine Quadratwurzeln aus negativen Zahlen existieren.
    Deshalb erweitern wir den Zahlenbereich auf die sogenannten
    komplexen Zahlen und lernen die imaginäre Einheit  $i$  kennen,
    für die gilt
20 EQUATION*
21  $i^2=-1$ .
22 Komplexe Zahlen werden oft mit  $z$  statt  $x$  notiert.
23
24 EXAMPLE @exa:intro
25 Um die Gleichung  $z^2=-9$  zu lösen, nutzen wir die Eigenschaft
     $i^2=-1$ :
26 EQUATION*
27  $z^2=-9=9 \cdot (-1)=9 \cdot i^2$ 
28 Jetzt kannst du die Quadratwurzel ziehen und erhältst die komplexe
    Zahl
29 EQUATION*
30  $z= \pm 3i$ 
31
32 PART
33 ICON=checkbox-marked-circle-outline
34
35 EXERCISE @ex:intro
36 CODE
37  $a = \text{rand}(4,9)$ 
38  $b = a^2$ 
39 Löse die Gleichung  $z^2=-b$ .
40  $z= \pm \sqrt{b}$ 
41

```

Abb. 4: Entwicklungsumgebung: Definition, Beispiel und randomisierte Aufgabe.

```

M u1-020-gausscheZahlenebene.mbl u1-030-konjGeo.mbl M X
hm1 > komplex > u1-030-konjGeo.mbl
1 Komplex Konjugierte in der Gaußschen Zahlenebene
2 #####
3
4 PART
5 ICON=help-circle-outline
6
7 EXERCISE @ex:konjGeo1
8 DISABLE_RETRY=true
9 CODE
10  $a = \text{rand}(5,7)$ 
11  $b = \text{rand}(2,4)$ 
12  $z = \text{complex}(a,b)$ 
13  $c = \text{conj}(z)$ 
14 FIGURE
15 WIDTH=75
16 PATH=img/u1-konjGeoEx1.svg
17 Du erhältst die komplexe Zahl  $z$ , wenn du die komplexe Zahl  $z$ 
18  $( )$  verdoppelst.
19  $( )$  am Ursprung spiegelst.
20  $(x)$  an der reellen Achse spiegelst.
21  $( )$  an der imaginären Achse spiegelst.
22
23 PART
24 ICON=book-open-page-variant-outline
25
26 DEFINITION @def:konjGeo
27 Du erhältst für die komplexe Zahl  $z=x+iy$  die komplex
    konjugierte Zahl  $\overline{z}$ , indem du  $z$  an der reellen
    Achse spiegelst:
28 EQUATION*

```

Abb. 5: Entwicklungsumgebung: Randomisierte Single Choice Aufgabe.

Ausblick

Inhaltlich sind neben Folgen und Reihen, Integralrechnung und komplexen Zahlen Teile der Linearen Algebra als weiteres Themengebiet geplant. Außerdem wurden bereits zusätzliche spielerische Elemente (Awards, Animationen, Minispiele) entworfen, die sich momentan in der Ausarbeitung befinden.

Förderung

Das Projekt wird von der Stiftung Innovation in der Hochschullehre (Freiraum 2022) gefördert.

Literaturverzeichnis

- [1] **Knospe, Heiko**: *Zehn Jahre Eingangstest Mathematik an Fachhochschulen in Nordrhein-Westfalen*, Proceedings zum 10. Workshop Mathematik in ingenieurwissenschaftlichen Studiengängen, Hochschule Ruhr-West, Mülheim an der Ruhr, 19–24 (2012).
- [2] **OMB+ Konsortium**: *Online Mathematik Brückenkurs Plus*, <https://ombplus.de> (2024).
- [3] **Kuzmin, Nikita and Ignatiev, Konstantin and Grafov, Denis**: *Experience of developing a mobile application using flutter*, Information Science and Applications: ICISA 2019, Springer, 571–575 (2020).
- [4] **Lister, Cameron and West, Joshua H and Cannon, Ben and Sax, Tyler and Brodegard, David and others**: *Just a fad? Gamification in health and fitness apps* JMIR serious games, JMIR Publications Inc., Toronto, Canada, 2, e3413 (2014).
- [5] **Saleem, Awaz Naaman and Noori, Narmin Mohammed and Ozdamli, Fezile**, *Gamification applications in E-learning: A literature review*. Technology, Knowledge and Learning, Springer, 27(1), 139–159 (2022).

Autoren

Patricia Graf, M.Sc.; Prof. Dr. Heiko Knospe; Andreas Schwenk, M.Sc.

TH Köln
 Betzdorfer Str. 2
 50679 Köln
 E-Mail: heiko.knospe@th-koeln.de

Malte Neugebauer

Lightweight Learning Analytics Dashboard for Analyzing the Impact of Feedback & Design on Learning in Mathematical E-Learning

Summary. Enriching mathematical digital exercises with approaches like gamification or pedagogical agents should support the learning processes of students. However, measuring the actual impact of such interventions requires the collection and sophisticated analysis of learning data, referred to as Learning Analytics. Because this mostly requires programming skills or is limited by institutional restrictions, often only performance is measured, which does not take learning as a process into account. To facilitate holistic analyses the present work introduces a lightweight Learning Analytics approach that can easily be implemented by non-programmers. Focusing feedback and design measurement methods for mathematical e-learning, a dashboard interface in the learning management system Moodle is implemented using this approach.

Introduction

Facing mathematics as a predictor for success in engineering studies on the one hand [6] and lower and lower scores in entry tests on the other [9], methods to enhance mathematical learning become desirable. Digital exercises haven proven to be practical for students regarding self-paced learning and interactive elements like immediate feedback as well as novel learning experiences such as game-based [15] or tutored learning [2].

However, the effects of such approaches are still the subject of current research. Gamification, e. g., is known to raise motivation [11], but the findings are mostly too context specific to be reproducible and the inquiry periods too short to speak of evidence beyond a novelty effect [4].

Learning Analytics could serve as a facilitator to gain more insight into the outcomes of such approaches in specific contexts. But despite the many studies on Learning Analytics, its implementation still seems to be a long way off [1]. Possible reasons for this are strict data protection regulations, long decision-making processes or inflexible IT infrastructures.

Thus, to investigate design and feedback related effects of digital exercises on a broad scale, a Learning Analytics approach would be nec-

essary that is easily implementable. Therefore, the main contribution of this work is the presentation of a solution to implement Learning Analytics methods easily into web-based learning management systems. To do so, studies with lightweight approaches from the field of design and feedback measurement are analyzed in the next section. Then a frontend-oriented Learning Analytics approach is derived from that and an example implementation in the learning management system Moodle is given. A concluding discussion points out the necessary future development.

Previous Work

Digital exercises' design [14] as well as automated feedback [3] have proven to be important factors for efficient learning processes. Even though many studies present interesting results for both of the areas mentioned, studies that enable their work to be easily reproduced either regarding the treatment method or the analysis are rare.

In a brief literature overview only two studies were found that offer the possibility to enrich mathematical digital exercises with design or feedback aspects in a lightweight way. [12] added simple non-interactive game elements to exercises and showed how marking exercises as “bosses” in combination with a difficulty-raising order of exercises leads to the most difficult tasks being repeated more often. With the help of error adaptive feedback this, in turn, led to a continuous increase of the learners' scores. [13] compared two experimental designs (pedagogical agent and gamification) with a control design and showed that the practice behavior of learners varies significantly among different designs. Both experimental designs foster the reattendance of exercises after not solving them correctly, whereas the gamified version additionally fosters repetition even after exercises were already solved correctly (i.e., practicing). Visualized in transition distribution charts (Figure 1) this results in higher values for repetition transitions (R) compared to sequential (S), non-sequential (N) or finish (F) transitions, independent of whether learners solved an exercise previously (T) correct (c), partially correct (p) or wrong (w).

The design aspects of these latter mentioned studies can be reproduced by pasting a script inside the texts of digital exercises in the learning management system. However, the analysis methods – although sufficiently described – are not part of the lightweight implementation which

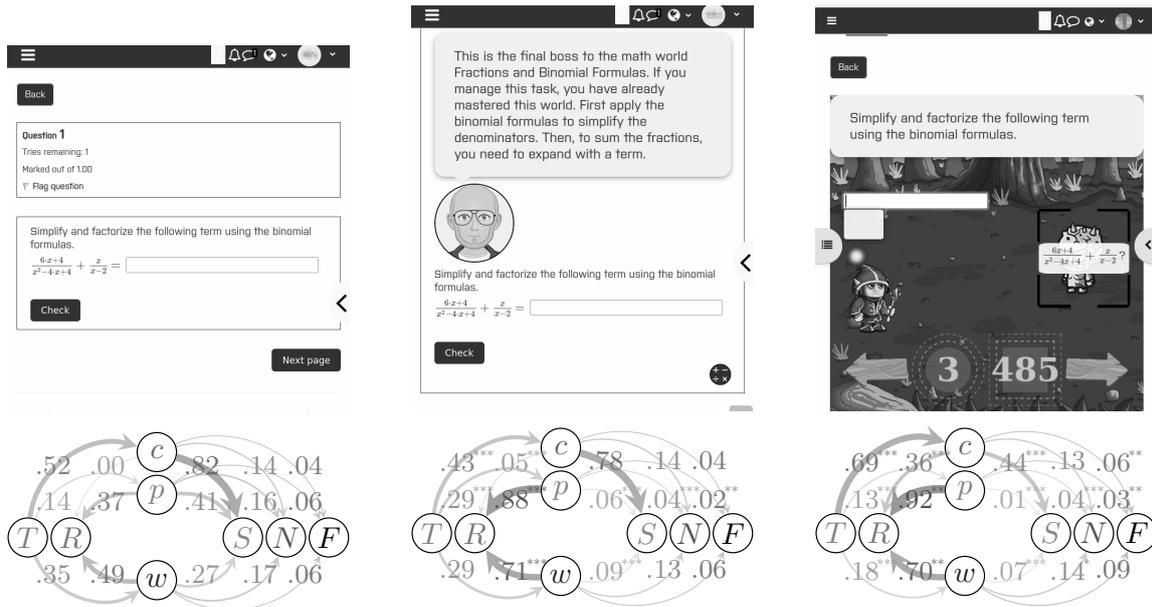


Figure 1: Different exercise designs and the according transitions charts [13]. Left to right: control, pedagogical agent, fantasy game.

makes it harder for researchers to build upon the findings. This gap can be closed with the solution described below.

Lightweight Approach & Example Implementation

Modern learning management systems offer course leaders interfaces that allow insight into learners' progression over time. Since this data is already displayed to users with the necessary rights, it can be gathered and visualized by making use of a frontend-oriented approach [12]. By interacting only with the frontend of the system (Figure 2) no server sided changes like the installation of a plugin are required. Researchers can, e.g., paste Javascript code directly into the browser's console. By doing this, a system that gathers the already available data and visualizes them according to Learning Analytics methods can be implemented.

An example implementation in the learning management system Moodle is given below. By gathering users' response history data, the analysis of feedback and design effects as presented in the previous section can be reconstructed and visualized (Figure 3). The *quiz dimension* (upper part) gives an overview of learners' interaction with the quiz as a whole. The upper left chart shows the transition distribution chart as

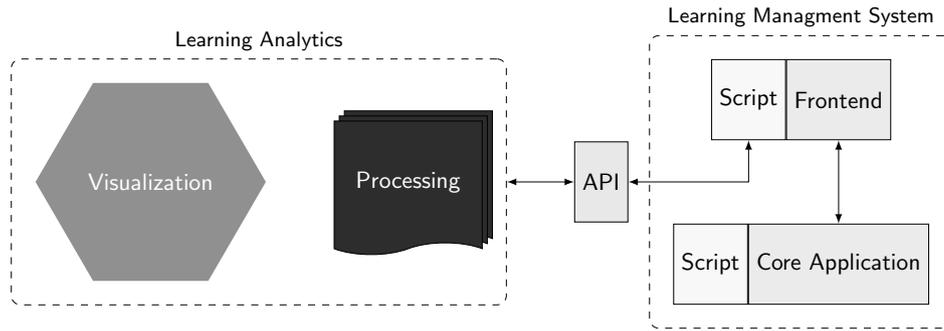


Figure 2: A frontend-oriented software architecture for Learning Analytics derived from [12].

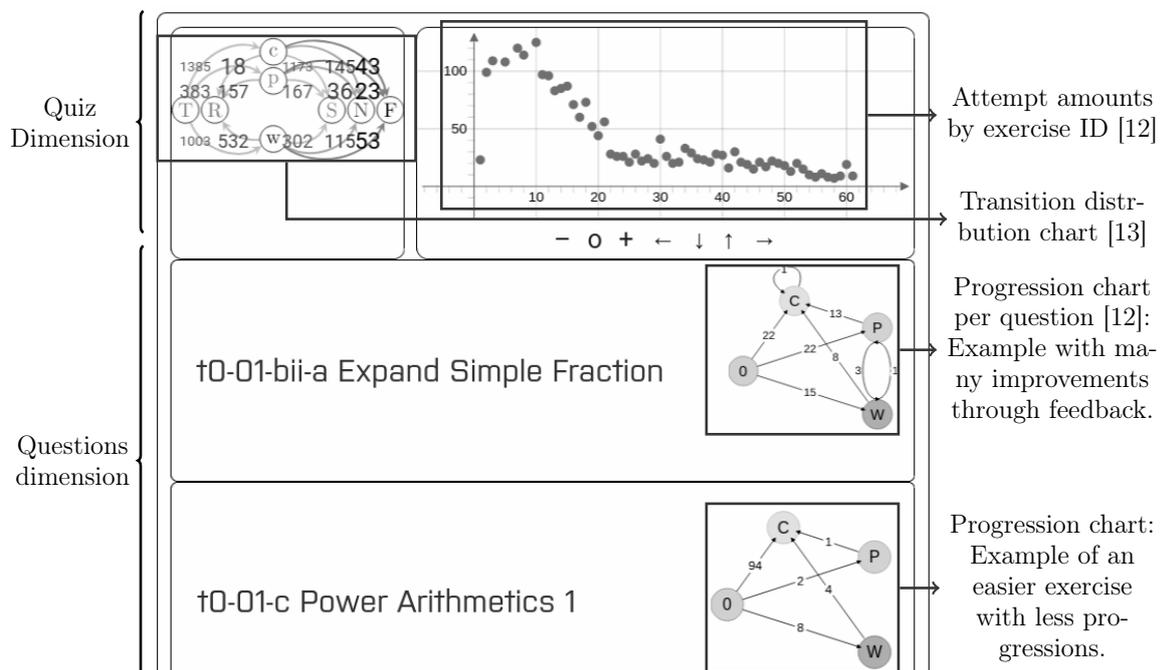


Figure 3: Screenshot of the dashboard.

introduced by [13] and described in the previous section. The upper right chart shows the amount of attempts for each question. Design-specific changes of, e.g., repetition rates can be visualized this way [12]. Each question is shown rowwise below. In this *questions dimension* next to the question name a progression chart is displayed, showing the movements of learners toward a correct answer (C) after initially (0) giving a wrong (W) or partially correct (P) answer. The amount of progressions toward better scores sheds light on the efficacy of specific feedbacks and allows question designers and researchers to adjust it accordingly [12].

The code for this example and a step by step guide for the implementation can be found at the project's repository (<https://bit.ly/3HRpyu0>).

Discussion & Conclusion

Aiming on improving mathematical teaching and learning, a feedback and design efficacy measurement dashboard for mathematical exercises was presented that can easily be implemented without restrictions. Known challenges in implementing Learning Analytics dashboards could be overcome by making use of a frontend-oriented software approach, that makes use of data that is already made available by the learning management system.

Previous studies show that the features presented by the dashboard are relevant indicators for efficient learning processes in the domain of mathematics. However, testing the dashboard with practitioners in future studies will shed light on the dashboard's benefits and pitfalls for both question designers' everyday use as well as for didactic research.

Bibliography

- [1] **Ahmad, A.; Schneider, J.; Weidlich, J.; Di Mitri, D.; Yau, J. Y.-K.; Schiffner, D.; Drachsler, H.:** *What Indicators Can I Serve You with? An Evaluation of a Research-Driven Learning Analytics Indicator Repository.* Proceedings of the 14th International Conference on Computer Supported Education - Volume 1: CSEDU 2022, **5**, 58-68(2022).
- [2] **Alrakhawi, H.; Jamiat, N.; Abu-Naser, S.:** *Intelligent Tutoring Systems in Education: A Systematic Review of Usage, Tools, Effects and Evaluation.* Journal of Theoretical and Applied Information Technology, 101, 1205-1226(2023).
- [3] **Cavalcanti, A. P.; Barbosa, A.; Carvalho, R.; Freitas, F.; Tsai, Y.-S.; Gašević, D.; Mello, R. F.:** *Automatic feedback in online learning environments: A systematic literature review.* Computers and Education: Artificial Intelligence, Elsevier BV, 2 (2021).
- [4] **Dichev, C.; Dicheva, D.:** *Gamifying education: what is known, what is believed and what remains uncertain: a critical review.* International Journal of Educational Technology in Higher Education, Springer Science and Business Media LLC, 14 (2017).
- [5] **Hattie, J.; Timperley, H.:** *The Power of Feedback.* Review of Educational Research, American Educational Research Association (AERA), 77, 81-112(2007).
- [6] **Heublein, U.; Ebert, J.; Hutzsch, C.; Isleib, S.; König, R.; Richter, J.:** *Zwischen Studierenerwartungen und Studienwirklichkeit, Ursachen des Studienabbruchs, beruflicher Verbleib der Studienabbrucherinnen und Studienabbrucher und Entwicklung der Studienabbruchquote an deutschen Hochschulen.* DZHW (2017).

- [7] **Knoop-van Campen, C.; Molenaar, I.:** *How Teachers integrate Dashboards into their Feedback Practices.* Frontline Learning Research, EARLI, 37-51(2020).
- [8] **Knoop-van Campen, C. A. N.; Wise, A.; Molenaar, I.:** *The equalizing effect of teacher dashboards on feedback in K-12 classrooms.* Interactive Learning Environments, Informa UK Limited, **31**, 3447-3463 (2021).
- [9] **Knospe, H.:** *Zehn Jahre Eingangstest Mathematik an Fachhochschulen in Nordrhein-Westfalen.* Proc. 10. Workshop Mathematik in ingenieurwissenschaftlichen Studiengängen, Mühlheim a.d. Ruhr, Hochschule Ruhr-West, 19-24(2012).
- [10] **Leeuwen, A.; Knoop-van Campen, C. A. N.; Molenaar, I.; Rummel, N.:** *How Teacher Characteristics Relate to How Teachers Use Dashboards: Results From Two Case Studies in K-12.* Journal of Learning Analytics, Society for Learning Analytics Research, **8**, 6-21(2021).
- [11] **Majuri, J.; Koivisto, J.; Hamari, J.:** *Gamification of Education and Learning: A Review of Empirical Literature.* Proceedings of the 2nd International Gam-iFIN conference, CEUR, 11-19(2018).
- [12] **Neugebauer, M.; Tousside, B.; Frochte, J.:** *Success Factors for Mathematical e-Learning Exercises Focusing First-Year Students.* Proceedings of the 15th International Conference on Computer Supported Education, SCITEPRESS - Science and Technology Publications 306-317(2023).
- [13] **Neugebauer, M.; Erlebach, R.; Kaufmann, C.; Mohr, J.; Frochte, J.** (in publication): *Efficient Learning Processes By Design: Analysis of Usage Patterns in Differently Designed Digital Self-Learning Environments.* Proceedings of the 16th International Conference on Computer Supported Education, SCITEPRESS - Science and Technology Publications (2024).
- [14] **Rienties, B.; Toetenel, L.:** *The impact of learning design on student behaviour, satisfaction and performance: A cross-institutional comparison across 151 modules.* Computers in Human Behavior, Elsevier BV, **60**, 333-341(2016).
- [15] **Saleem, A. N.; Noori, N. M.; Ozdamli, F.:** *Gamification Applications in E-learning: A Literature Review.* Technology, Knowledge and Learning, Springer Science and Business Media LLC, **27**, 139-159(2021).

Author

Malte Neugebauer
 University of Applied Sciences Bochum
 Campus Velbert-Heiligenhaus
 Kettwiger Str. 20
 D-42579 Heiligenhaus
 E-Mail: malte.neugebauer@hs-bochum.de

Hildo Bijl

Application of Performance Distribution Tracing to a Thermodynamics Course

Abstract. Online education is rapidly growing, but many courses struggle with how to detect skill mastery. This paper presents the application of the Performance Distribution Tracing (PDT) algorithm to a university-level thermodynamics course. In this experiment PDT provided live tracking of the estimated probability of success of students on 35 course skills. This probability is shown as ‘skill level’ feedback to students. On top of that the estimations are also used to provide recommendations to students on which skill to practice. Students reported a more structured study process, an increase in study motivation and a reduction of stress.

Introduction

With the advent of online education, a multitude of online courses is appearing on the web. Many of these courses also have accompanying exercises, partly for students to practice, and partly to see whether or not they have obtained the relevant skills. This brings up the main question, ‘How can we determine if a student has sufficiently mastered a given skill?’ This question needs to be answered to determine if the student may and/or should continue to the next part of the course.

Various e-learning companies tackle this in different ways. Common methods in practice include:

- Make a test of n (possibly randomized) exercises and score at least $p\%$. On a failure, do a new (possibly randomized) test.
- Make a test of n (possibly randomized) exercises. If you get any incorrect, repeat those exercises until everything is correct.
- Do single randomized exercises until you get m in a row correct.
- A single failure already gets you sent back to the theory pages.
- And dozens of other variations.

Each of these methods has downsides. Students that already have sufficiently mastered the content matter often have to do a lot of work to prove said mastery, especially if they also make occasional errors, whereas students who do not understand the subject at all may simply repeat

exercises until they learn the ‘trick’, while still failing to grasp the content matter. As a result, a skill tracing method is needed that quickly detects mastery, while also recognizing consistent failure.

In this paper, a novel skill tracing method – Performance Distribution Tracing (PDT) – is applied to a basic thermodynamics course. To see how this works, we first study and compare existing knowledge/skill tracing methods, briefly explaining the idea of PDT. This PDT algorithm is then applied in a basic thermodynamics course, after which the user experiences are evaluated, both with the skill tracing and with the accompanying recommendations. The paper closes off with conclusions and recommendations.

Existing knowledge/skill tracing algorithms

It is common in education to define so-called Knowledge Components (KCs), either explicit KCs (facts/principles) or implicit KCs (skills) [7]. The probability that a student performs an implicit KC successfully is called the success rate, and tracking/estimating it is known as Knowledge Tracing (KT) [16, 11].

Classical work revolves around Bayesian Knowledge Tracing (BKT) to directly estimate the success rate [3], taking into account a variety of factors like guessing and slipping [1, 12, 22, 21]. Other methods use logistic models [2, 13, 18]. These methods all relate to individual skills, not taking into account domain knowledge like links between skills. This domain knowledge can be learned internally [14, 5, 20] through deep learning, but this requires copious amounts of data that is often not available. It can also be applied externally through Bayesian networks [10, 8, 15, 4, 9, 17], but (on top of often being computationally intensive) these methods do not provide any data on the accuracy of its estimates. This can be fixed by analytically tracing the distribution (through the probability density) of the success rate. That is the main idea behind the Performance Distribution Tracing (PDT) algorithm applied in this paper.

The performance distribution tracing algorithm

The Performance Distribution Tracing (PDT) algorithm assumes there is some probability \underline{x} that a student will apply a certain skill correctly. Its

essence is that it treats this probability as a random variable and approximates its distribution through beta-distribution $\beta(x)$ basis functions (see for instance [6, chapter 25]), according to

$$f_{\underline{x}}(x) = \sum_{i=0}^n c_i (n+1) \binom{n}{i} x^i (1-x)^{n-i} = \sum_{i=0}^n \beta_{i+1, n-i+1}(x). \quad (1)$$

Whenever a new observation is made – a student performs an exercise correctly or incorrectly – the order n and coefficients c_0, \dots, c_n of this probability density function are updated.

The PDT algorithm has various strengths. It can apply updates with relatively little computational effort. It can apply updates for convoluted observations; for instance a student does an exercise requiring multiple different skills. It can take into account links and/or hierarchy between related skills. And it provides a distribution as a result, meaning that the PDT algorithm also shows how certain it is about its predictions.

Set-up of the experiment in the Step-Wise app

To evaluate the PDT algorithm in practice, it has been tried out within a university-level thermodynamics course. The course has been split up into 35 skills, many with complex interdependencies. The students were taught about the theory in regular lectures. They were then instructed to practice using the Step-Wise app (see Step-Wise.com). An example exercise is shown in Figure 1.

The PDT algorithm provides the students with continuous information about their skill levels; see the skill sphere in Figure 1. It also makes suggestions, based on this skill level information, on what the student can best practice at the current moment. On top of this, students have the ability to split any exercise up into steps. These steps are then connected to skills that the student has previously shown to have mastered already. This allows the student to get more familiar with how to solving exercises, while at the same time providing the PDT algorithm with valuable information on which subskill the student might be struggling with.

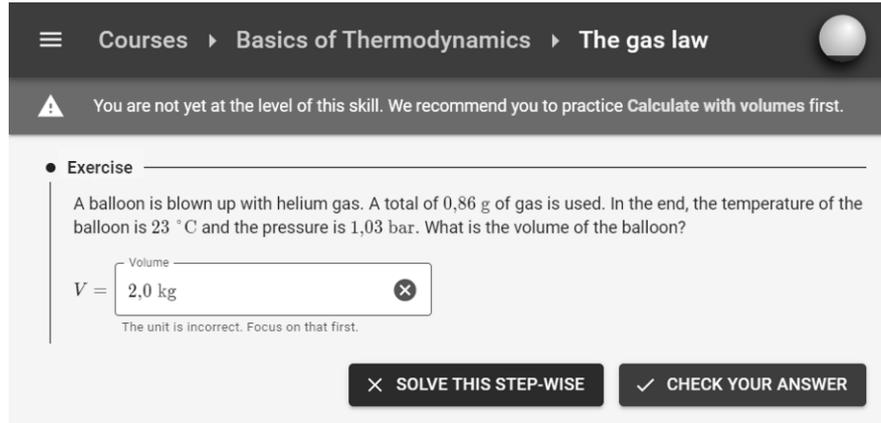


Figure 1: A screenshot of an exercise from the Step-Wise app. Numbers are randomly generated. Every exercise is connected to a skill, and the success rate estimate is continuously shown as a colored sphere on the top right. By hovering over the sphere, the student sees an explanation of what this score is and is based on. Exercises are programmed to provide individualized feedback.

Student experiences with success rate estimates

The main goal of PDT was to prevent counterproductive situations: students with mastery who have to continue practicing, or students without mastery who can continue. Interviews with students have shown that this generally was the case. Students who master the subject well mentioned they went through everything quickly, while students who did not master the topic matter had to keep repeating exercises for the same skill, as is desired. A few students mentioned, ‘On a silly calculation error the app already counts it as a fail, dropping my carefully accumulated score. On an exam I do get partial points for a correct method!’ Since working accurately and checking your own work is part of the learning goals, this shows the system works as intended. And indeed, the students involved also reported checking their work more for errors.

On top of the above result, students mentioned an increase in motivation due to the skill level feedback. One student said, ‘I really wanted the indicator to turn green. I worked deep into the night to get there.’ So this type of feedback can increase the time spent on task. Next to that another student said, ‘It is very reassuring that the app tells me that I have practiced enough.’ This manner of skill level feedback can hence also reduce stress.

Experiences of skill and exercises recommendations

The skill level information was not only used to provide feedback to students, but also to give practice recommendations of the type ‘It would be best if you now practice [some skill].’ In the thermodynamics course, the students mostly liked these recommendations since it structured their learning process, and they eagerly and obediently followed them. Over time this did decrease, as students got more familiar with the course materials and had their own feeling of ‘I want to practice [other skill] some more.’ Students were happy that the recommendations were merely recommendations, and they could still follow their own plan if desired.

The Step-Wise app also includes an intelligent exercise selection method, roughly similar to [19]. On top of estimating skill success rates, the PDT algorithm is capable of estimating success probabilities for exercises too. (Note that some exercises are more complicated, for instance have more steps, than others.) The Step-Wise app was programmed to more likely select exercises whose estimated success rate was not too low (too easy) but also not too high (too hard). This add-on worked well – students started with easier exercises and later on moved on to harder ones – but it had an unexpected negative downside. Due to the limited number of exercises available, struggling students often kept getting the same exercise, which at times caused minor frustrations. Adding more exercises would solve this problem.

Conclusion and recommendations

In general it can be concluded that the PDT algorithm works well in practice. Students reported that the skill level feedback increased their motivation as well as reduced stress levels. On top of this, the skill level data can be used for practice recommendation and exercise selection, making the study process more effective.

Naturally there are always further improvements to make. Part of these improvements focus on the practical application. For instance, the Step-Wise app would benefit greatly from a larger exercise set. Another part focuses on the PDT algorithm itself: it currently does not include student-specific data. In theory PDT could detect that some students pick up skills more quickly than others and adjust its updates based on this. This individualization idea is left for further research.

Bibliography

- [1] **Baker, R. S. J. D., Corbett, A. T., and Aleven, V.:** *More Accurate Student Modeling through Contextual Estimation of Slip and Guess Probabilities in Bayesian Knowledge Tracing*. Intelligent Tutoring Systems, 406-415 (2008).
- [2] **Cen, H., Koedinger, K., and Junker, B.:** *Learning Factors Analysis – A General Method for Cognitive Model Evaluation and Improvement*. Intelligent Tutoring Systems, 164–175 (2006).
- [3] **Corbett, A. T. and Anderson, J. R.:** *Modeling the acquisition of procedural knowledge*. User Modeling and User-Adapted Interaction 4, 253–278 (1994).
- [4] **Cui, Y., Chu, M.-W., and Chen, F.:** *Analyzing Student Process Data in Game-Based Assessments with Bayesian Knowledge Tracing and Dynamic Bayesian Networks*. Journal of Educational Data Mining 11, 1 (06), 80–100 (2019).
- [5] **Guo, X., Huang, Z., Gao, J., Shang, M., Shu, M., and Sun, J.:** *Enhancing Knowledge Tracing via Adversarial Training*. Association for Computing Machinery, New York, NY, USA, 367–375 (2021).
- [6] **Johnson, N. L., Kotz, S., and Balakrishnan, N.:** *Continuous Univariate Distributions*. Wiley Series in Probability and Statistics, Volume 2 (1995).
- [7] **Koedinger, K., Corbett, A., and Perfetti, C.:** *The Knowledge-Learning-Instruction (KLI) Framework: Toward Bridging the Science-Practice Chasm to Enhance Robust Student Learning*. Cognitive Science 36, 757-798 (2012).
- [8] **Käser, T., Klingler, S., Schwing, A. G., and Gross, M.:** *Dynamic Bayesian Networks for Student Modeling*. IEEE Transactions on Learning Technologies 10, 4, 450–462 (2017).
- [9] **Levy, R.:** *Dynamic Bayesian Network Modeling of Game-Based Diagnostic Assessments*. Multivariate Behavioral Research 54, 6, 771–794 (2019).
- [10] **Levy, R. and Mislevy, R. J.:** *Bayesian Psychometric Modeling*. Chapman & Hall (2016).
- [11] **Liu, Q., Shen, S., Huang, Z., Chen, E., and Zheng, Y.:** *A Survey of Knowledge Tracing*. (2021).
- [12] **Pardos, Z. A. and Heffernan, N. T.:** *KT-IDEM: Introducing Item Difficulty to the Knowledge Tracing Model*. User Modeling, Adaption and Personalization – 19th International Conference (2011).
- [13] **Pavlik Jr, P., Cen, H., and Koedinger, K.:** *Performance Factors Analysis - A New Alternative to Knowledge Tracing*. Frontiers in Artificial Intelligence and Applications 200, 531–538 (2009).
- [14] **Piech, C., Bassen, J., Huang, J., Ganguli, S., Sahami, M., Guibas, L. J., and Sohl-Dickstein, J.:** *Deep Knowledge Tracing*. Advances in Neural Information Processing Systems 28 (2015).

- [15] **Reichenberg, R. E.:** *Dynamic Bayesian Networks in Educational Measurement: Reviewing and Advancing the State of the Field*. Applied Measurement in Education 31, 4 (10), 335–350 (2018).
- [16] **Sapountzi, A., Bhulai, S., Cornelisz, I., and Klaveren, C.:** *Dynamic Knowledge Tracing Models for Large-Scale Adaptive Learning Environments*. International Journal on Advances in Intelligent Systems 12, 93 – 110 (2019).
- [17] **Sheidaei, A., Foroushani, A. R., Gohari, K., and Zeraati, H.:** *A novel dynamic Bayesian network approach for data mining and survival data analysis*. BMC Medical Informatics and Decision Making 22, 1 (2022).
- [18] **Vie, J.-J. and Kashima, H.:** *Knowledge Tracing Machines: Factorization Machines for Knowledge Tracing*. Proceedings of the AAAI Conference on Artificial Intelligence 33, 750–757 (2019).
- [19] **Wu, Z., Li, M., Tang, Y., and Liang, Q.:** *Exercise recommendation based on knowledge concept prediction*. Knowledge-Based Systems 210 (2021).
- [20] **Yang, S., Zhu, M., and Lu, X.:** *Deep Knowledge Tracing with Learning Curves*. (2021).
- [21] **Yudelson, M.:** *Individualization of Bayesian Knowledge Tracing Through Elo-infusion*. Lecture Notes in Computer Science 12749, 412-416 (2021).
- [22] **Yudelson, M. V., Koedinger, K. R., and Gordon, G. J.:** *Individualized Bayesian Knowledge Tracing Models*. Lecture Notes in Computer Science 7926, 171-180 (2013).

Author

Dr. Hildo Bijl
Hochschule Bochum
Am Hochschulcampus 1
44801 Bochum
E-Mail: hildo.bijl@hs-bochum.de

NOTIZEN

NOTIZEN

Anhang

WFR - Wismarer Frege-Reihe / Wismar Frege Series

Beiträge zur Mathematikausbildung von Ingenieuren

- Heft 01/2005 Proceedings 4. Workshop Mathematik für Ingenieure, Bremen, Oktober 2005.
- Heft 05/2006 Proceedings 5. Workshop Mathematik für Ingenieure, Wismar, Teile 1 – 3, September 2006.
- Heft 01/2007 Proceedings Minisymposium Moderne Mathematikausbildung für Ingenieure, Humboldt-Universität Berlin, Teile 1 – 2, März 2007.
- Heft 02/2007 Mathematik für Ingenieure – Thesen zum Jahr der Mathematik 2008, Dezember 2007.
Mathematics for Engineers – Theses to the Year of Mathematics 2008, December 2007.
- Heft 03/2008 Proceedings 6. Workshop Mathematik für Ingenieure, Soest, Teile 1 – 2, September 2008.
- Heft 04/2008 Proceedings Minisymposium Moderne Mathematikausbildung für Ingenieure, Friedrich-Alexander-Universität Erlangen-Nürnberg, Teile 1 – 2, September 2008.
- Heft 03/2009 Peter Junglas: Interaktive Simulationsprogramme zur Demonstration von klassischen und quantentheoretischen Wellenphänomenen, Juni 2009.
- Heft 04/2009 Proceedings 7. Workshop Mathematik für Ingenieure, Wolfenbüttel, Juni 2009.
- Heft 02/2010 Information – Programme and Abstracts, 15th SEFI MWG Seminar & 8th Workshop GFC, Wismar, June 2010.
- Heft 03/2010 Proceedings 8. Workshop Mathematik für Ingenieure, Wismar, Juni 2010.
- Heft 05/2010 Larissa Fradkin: Teaching Algebra and Calculus to Engineering Entrants, December 2010.
- Heft 02/2011 Proceedings 9. Workshop Mathematik für ingenieurwissenschaftliche Studiengänge, Wilhelmshaven, September 2011.
- Heft 03/2013 Proceedings 11. Workshop Mathematik für ingenieurwissenschaftliche Studiengänge, Teile 1 – 2, Bochum, September 2013.

Heft 02/2015	Proceedings 12. Workshop Mathematik für Ingenieure, Teile 1 – 2, Hamburg, Februar 2015.
Heft 04/2016	Proceedings 13. Workshop Mathematik in ingenieurwissenschaftlichen Studiengängen, Lingen, September 2016.
Heft 01/2017	Proceedings 14. Workshop Mathematik in ingenieurwissenschaftlichen Studiengängen, Erlangen, September 2017.
Heft 01/2018	Sergiy Klymchuk: Puzzle-Based Learning in Engineering Mathematics: Students' Attitudes.
Heft 02/2018	Proceedings 1st Northern-Light Symposium on Engineering Education, Hamburg, April 2018.
Heft 02/2019	Proceedings 15. Workshop Mathematik in ingenieurwissenschaftlichen Studiengängen, Rostock-Warnemünde, April 2019.
Heft 03/2019	Proceedings 2nd Northern-Light Symposium on Mathematical Education in Engineering, Hamburg, September 2019.
Heft 02/2020	Proceedings 16. Workshop Mathematik in ingenieurwissenschaftlichen Studiengängen, Dortmund, Mai 2020.
Heft 01/2022	Das Mathematikabitur von Gottlob Frege. Dezember 2020.
Heft 03/2022	Proceedings 3rd Northern-Light Symposium on Mathematical Education in Engineering, Wismar, September 2022.
Heft 01/2023	Proceedings 4th Northern-Light Symposium on Mathematical Education in Engineering, Hamburg, September 2023.
Heft 02/2023	Proceedings 18. Workshop Mathematik in ingenieurwissenschaftlichen Studiengängen, Bochum, November 2023.

Hinweise:

- Die Proceedings zur Workshop-Reihe beginnen erst mit dem 4. Workshop.
- Die Proceedings zum 10. Workshop erschienen in einem Extraband an der Hochschule Ruhr/West in Mülheim.
- Die Proceedings zum 17. Workshop sind bisher nicht erschienen.

Herausgeber und Redakteur

Prof. Dr. rer. nat. habil. Dieter Schott
Gottlob-Frege-Zentrum
Fakultät für Ingenieurwissenschaften
Hochschule Wismar
Philipp-Müller-Str. 14
D - 23966 Wismar
Telefon: ++49 / (0)3841 / 753 7333
Fax: ++49 / (0)3841 / 753 7130
E-Mail: dieter.schott@hs-wismar.de

Vertrieb:

Direkt über den Herausgeber oder das Gottlob-Frege-Zentrum

ISSN 1862-1767
ISBN 978-3-947929-25-2