

# Aufwand und Nutzen parametrisierbarer, pfadbasierter Aufgaben

Nils Schwinning<sup>1</sup>, Melanie Schypula<sup>1</sup>, Michael Striewe<sup>1</sup> und Michael Goedicke<sup>1</sup>

**Abstract:** Parametrisierbare, pfadbasierte Aufgaben sind ein Konzept, mit dem einfache Fragentypen wie Multiple-Choice und Fill-In flexibilisiert und zu komplexen dynamischen Aufgaben zusammengesetzt werden können. Mit den Möglichkeiten des Aufgabendesigns steigt auch der Aufwand für die Erstellung der Aufgaben. Dieser Beitrag stellt ein Konzept für die Umsetzung solcher Aufgaben vor und diskutiert den didaktischen Nutzen und den Aufwand zur Erstellung der Aufgaben.

**Keywords:** E-Assessment, Adaptives E-Learning, Feedbacksysteme

## 1 Einleitung

In großen Lehrveranstaltungen ist es oft unmöglich, allen Studierenden individuelle Feedbacks zu Lösungen von Übungsaufgaben zu geben. Abhilfe versprechen seit einigen Jahren adaptive E-Assessment-Systeme, durch die den Lernenden passende Aufgaben angezeigt werden. Ansätze finden sich im Projekt ACTIVEMATH [Gog11] und im QTI-Standard<sup>2</sup>, wobei die Randomisierung in beiden Fällen auf bestimmte Datentypen und Auswahlmechanismen beschränkt ist. Flexibler sind unabhängige Komponenten zur Einbindung in E-Learning-Systeme, wie man sie bei WIRIS<sup>3</sup> findet, dessen „quiz engine“ und Computer-Algebra-System in MOODLE genutzt werden kann [HA14]; oder ein Softwarepaket für die Statistik-Programmiersprache R, mit dem aus Vorlagendateien Aufgaben mit randomisierten Inhalten generiert werden können, die in verschiedenen Formaten (z.B. für die Learning-Management-Systeme (LMS) MOODLE und OLAT oder für den QTI-Standard) exportiert werden können. In beiden Fällen kann sich jedoch das umgebende LMS als limitierender Faktor für die mögliche Komplexität von Aufgaben und Feedback erweisen.

Um dieser Schwierigkeit zu begegnen wurde für das E-Assessment-System JACK ein neuer Aufgabentyp entwickelt, mit dem für Multiple-Choice und Fill-In Fragen sowohl individuelle automatische Feedbacks generiert, als auch auf die Eingaben der Studierenden reagiert und die Aufgabe daran angepasst werden kann. Weiterhin kann dieser Aufgabentyp variable Bestandteile besitzen, wodurch das System eine große Anzahl ähnlicher, aber dennoch verschiedener Aufgaben automatisch generieren kann [SSSG14]. Natürlich sind derlei Aufgaben in der Erstellung komplexer als statische Aufgaben. Es

---

<sup>1</sup>Universität Duisburg-Essen, Gerlingstraße 16, 45127 Essen, vorname.nachname@s3.uni-due.de

<sup>2</sup><http://www.imsglobal.org/question/>

<sup>3</sup><http://www.wiris.com>

ist daher zu untersuchen, welcher Mehraufwand für die Erstellung derartiger Fragen notwendig ist und welcher didaktische Nutzen oder sonstiger Mehrwert aus solchen Aufgaben gewonnen werden kann.

## 2 Pfadbasierte Aufgaben in JACK

Eine pfadbasierte Aufgabe in JACK ist ein gerichteter Graph, die einzelnen Knoten des Graphen sind Teilaufgaben, sogenannte Stufen. Eine Stufe ist eine in sich geschlossene Aufgabe vom Typ Multiple-Choice, Fill-In oder Wahr/Falsch. Die verschiedenen Stufen einer Aufgabe können durch Bedingungen miteinander verknüpft werden. So kann das System bspw. bei einer richtigen Lösung zu einer anderen Stufe wechseln als bei einer falschen Lösung. Dadurch erlauben die pfadbasierten Aufgaben ein hohes Maß an Adaptivität. Zudem können Aufgaben in JACK mittels Variablen parametrisiert werden, so dass bei jeder Darstellung der Aufgabe eine für die Studierenden individuelle Variante generiert wird. Eine Variable besteht aus einem Namen, einem Typ und einer Beschreibung zur Bestimmung ihres Wertes. In Variablen können einzelne Elemente wie natürliche Zahlen, reelle Zahlen oder Zeichenketten gespeichert werden oder Mengen, die verwendet werden können, um weitere Variablen zufällig zu ziehen.

In den Aufgabentext kann der Inhalt von Variablen und somit auch die bereits getätigten Eingaben der Studierenden eingebunden werden. Dazu kann entweder direkt der Inhalt der Variablen verwendet werden oder ein Ausdruck in den Text eingefügt werden, der eine Berechnung auf den Variablen ausführt. Bei parametrisierten Aufgaben müssen die Variablen, mit denen die Parametrisierung vorgenommen wurde, natürlich auch mit in die Überprüfung der Ergebnisse einfließen. Das Konzept pfadbasierter Aufgaben sieht zusätzlich vor, dass mit den Eingaben von Studierenden aus einer Stufe in allen Folgestufen weiter gearbeitet werden kann. Diese Möglichkeit erweist sich besonders bei divergenten Aufgaben, Aufgaben mit unendlich vielen richtigen Lösungen, als sinnvoll.

## 3 Anwendungserfahrungen

Die formularbasierten Aufgaben werden an der Fakultät für Wirtschaftswissenschaften der Universität Duisburg-Essen in diversen Vorlesungen mit großen Teilnehmerzahlen eingesetzt. Im Wintersemester 2013/14 bzw. Sommersemester 2014 kamen sie in je einer Informatik-Veranstaltung des Bachelor- und Masterstudiums zur Prüfungsvorbereitung zum Einsatz mit 9 Aufgaben. Zudem wurde JACK in den Vorlesungen “Deskriptive Statistik” (137 Aufgaben), “Induktive Statistik” (175 Aufgaben) und “Mikroökonomik I-III” (96 Aufgaben) der Studiengänge Betriebswirtschaftslehre und Volkswirtschaftslehre eingesetzt. Das Konzept dieser Veranstaltungen beinhaltet vorlesungsbegleitende Übungen und Testate mit JACK. In der Vorlesung “Induktive Statistik” wurde zusätzlich eine elektronische Klausur mit JACK angeboten. Des Weiteren gibt es einen Vorkurs für Mathematik mit 300 Aufgaben, welcher das System verwendet.

Über 80% dieser Aufgaben sind variabel.

Lehrende, die pfadbasierte und randomisierte Aufgaben erstellen möchten, müssen einen erheblichen Mehraufwand gegenüber einstufigen, nicht randomisierten Aufgaben leisten. Zunächst müssen sich die Lehrenden überlegen, welche Teile einer Aufgabe randomisiert werden können. Bei Multiple-Choice Aufgaben können die Disktraktoren randomisiert werden, bei Lückentextaufgaben bestimmte Zahlenwerte oder Wörter in der Aufgabenstellung. Ist letzteres der Fall, so kann auch die richtige Lösung abhängig von den Variablen sein. So ist es aus didaktischer Sicht nicht sinnvoll, zu große Zahlenwerte in der Aufgabenstellung oder „krumme“ Zahlenwerte als Lösung zu haben. Es kann vorkommen, dass es von der Wahl der Variablen abhängt, ob eine Aufgabe eine Lösung besitzt oder nicht. Werden die Disktraktoren einer Multiple-Choice Aufgabe zufällig aus einer Menge gezogen, so muss sichergestellt werden, dass nicht zwei Mal das gleiche Element gezogen wird. Falls verschiedene Pfade durch die Aufgabe führen sollen, müssen Lehrende sich beim Erstellen der Aufgabe genau überlegen, welche Fehler zu erwarten sind und wie sie mit einer Folgestufe darauf reagieren möchten. Lernende, die keinen Lösungsansatz haben, neigen z.T. zu Eingaben wie „keine Ahnung“ oder „weiß nicht“, die im Kontext der Aufgabe nicht korrekt sein können.

Es sollten alle Pfade einer pfadbasierten, randomisierten Aufgabe durchlaufen, alle Feedbackfälle ausprobiert und die Variablenkonfiguration stichprobenartig überprüft werden, bevor sie in den Produktivbetrieb gehen kann. Der zeitliche Aufwand, welcher sich für das Testen aller möglichen Fälle ergibt, ist für Lehrende nicht vollständig zu leisten. Befragungen der Lehrenden haben ergeben, dass 4-8 Stunden benötigt werden, um eine randomisierte, pfadbasierte Aufgabe zu erstellen. Dem gegenüber steht eine hohe Zeitersparnis, wenn die Aufgabe hinreichend viele Alternativen anbietet und somit keine nahezu identischen zusätzlichen Aufgaben erstellt werden müssen. Da es trotz des hohen Aufwands eine große Anzahl dieser Aufgaben gibt bestätigt, dass die Lehrenden diesen Aufwand als lohnenswert einschätzen. Mit der Fertigstellung eines funktionsfähigen Editors ist eine Reduktion des Aufwands zu erwarten.

## Literaturverzeichnis

- [Gog11] Goguadze, G.: Active Math – Generation and Reuse of Interactive Exercises using Domain Reasoners and Automated Tutorial Strategies. Dissertation, Universität des Saarlandes, 2011.
- [HA14] Hofrichter, R.; Aguiló, C.: Der Einsatz von WIRIS quizzes im LMS Moodle für Online-Self-Assessments im Fach Mathematik an der Hochschule Pforzheim. In DeLFI 2014 – Die 12. e-Learning Fachtagung Informatik, Bonn, 2014. GI.
- [SSSG14] Schwinning, N.; Schypula, M.; Striewe, M.; Goedicke, M.: Concepts and Realisations of Flexible Exercise Design and Feedback Generation in an e-Assessment System for Mathematics. In Joint Proceedings of the MathUI, Open-Math and ThEdu Workshops and Work in Progress track at CICM, 2014.