Mike Altieri, Jörg Horst, Michael Kallweit, Karin Landenfeld, Malte Persike (2020). Multi-step procedures in STACK tasks with adaptive flow control. In: Contributions to the 3rd International STACK Conference 2020. TTK University of Applied Sciences: Tallinn, Estonia.
DOI: https://doi.org/10.5281/zenodo.3944786

Multi-step procedures in STACK tasks with adaptive flow control

Mike Altieri[1], Jörg Horst[2], Michael Kallweit[3], Karin Landenfeld[4], Malte Persike[5]

Abstract: Digital exercises are increasingly becoming a mainstay of mathematics and engineering lectures worldwide. The time when digital tasks were only used to realize analog computational tasks with a schematic solution process and a check of the final result is long gone. In the meantime, didactically valuable and challenging tasks are being designed, with STACK taking a leading role in their realization. The further developments of STACK are correspondingly diverse and dynamic. One such further development, which is not yet part of the current STACK release, is the adaptive flow control of the solution process depending on the intermediate results in complex tasks. In this paper the technical realization is discussed, the added value is illustrated by two examples and finally an outlook on future planning is given.

Keywords: STACK, adaptivity, flow control, GeoGebra, JavaScript

1 Introduction

The introduction of modern technologies into university teaching has opened up new possibilities for designing digitally supported teaching concepts. An important component of these are digital exercises, as they can be realized in STACK. Here, randomized tasks can be constructed, with which students can practice online as often as they like [Sa13]. Didactically challenging tasks can be developed, ranging from the integration of dynamic geometry [Kl19], adaptive feedback [We19] and the integration of interactive H5P videos [Al19] to *external* adaptive flow control [Br17], in which learners are guided to easier or more difficult tasks depending on their results. However, the possibility of reacting adaptively to incorrect intermediate results *within* tasks has not yet been sufficiently considered. The current STACK release still lacks the technical prerequisites for implementing this feature, which we call *internal* adaptive flow control. For this reason, it was implemented in a not yet official STACK branch. The following section 2 describes moodle's capabilities for adaptivity and our implementation. Section 3 contains two examples from linear algebra and analysis. Section 4 summarizes the current status and future plans.

1 Ostbayerische Technische Hochschule Amberg-Weiden, m.altieri@oth-aw.de
2 Bielefeld University Of Applied Sciences, joerg.horst@fh-bielefeld.de
3 Ruhr-Universität Bochum, michael.kallweit@rub.de
4 Hamburg University of Applied Sciences, karin.landenfeld@haw-hamburg.de
5 RWTH Aachen, persike@cls.rwth-aachen.de

2 Capabilities for adaptivity in Moodle and our approach

Moodle is one of the most used Learning Management Systems worldwide. Besides many other functionalities, it also offers some possibilities for adaptivity, for example:

- Course activities, like Adaptive Quiz or Simple lesson. Adaptive tests consist of questions selected from the question bank that are tagged with a score of their difficulty. the next question is selected based on the current score of the user. The lesson module presents a series of pages to the student who is usually asked to make some sort of choice. The choice will send them to a specific page in the Lesson.
- Course settings, like the restricted access, which enables teachers to restrict the availability of any activity according to certain conditions.
- The possibility of making the opening of questions conditional depend upon the completion of a previous question.

All these solutions work on the level of what we call *outer* adaptivity. That is the adaptivity of the sequence of questions. Every question is a fresh start and none of the information we gathered in previous questions (i.e. the scores of users) can be incorporated into next ones. To achieve this, one approach is to make the question change itself. At first it presents the main exercise, but on the way of answering it maybe changes to intermediate step exercises.

We call this *inner* adaptivity and introduced it at the level of individual STACK questions [Ka19]. For each question, the potential response tree analyses the students' solution, and if they have made an error, shows a link to the starting point of an individual path. For example, the next intermediate step exercise becomes visible only by clicking on a button. The analysis of the input for this intermediate step is stored in a separate response tree which adaptively defines the subsequent tasks.

We created an easy-to-use solution:

1. Include an externally stored JavaScript file in your question text. The script provides functions to show and hide sections immediately or after a delay or even only if a condition is satisfied.
2. Create the sections of the questions you would like to control with separate HTML div-containers.
3. Add calls to the new functions in the HTML source of text fields, for example one can add a button in a feedback text, which shows the next section on its onclick-event.

To use the inner adaptivity of questions in a moodle quiz, one must set the question behaviour to “interactive”, so that every question and their build-in intermediate step exercises are evaluated immediately.

3 STACK tasks with adaptive flow control

3.1 Example 1: Adaptive flow control in a combined STACK + GeoGebra task

An important task in linear algebra is the calculation of eigenvalues and eigenspaces of a matrix. In the following example, the lines are to be shifted so that they correspond to the eigenspaces of the matrix in Fig. 1. If the lines have been shifted correctly, the learners will receive positive feedback. If the lines are shifted in a wrong way, the learners could previously only be told that the result is incorrect. Hence, until now it was not possible to determine where within the calculation an error occurred, one could only check for typical errors in the final result. In multi-step procedures this results in a check of each calculation step what is very time-consuming. In addition, in this case a long search for errors significantly reduces the user experience when processing the digital task. Thanks to the internal adaptive flow control it is now possible to easily identify the error on the way to the solution: First, it is determined at which point an incorrect intermediate result occured, in order to then ask for a recalculation of the intermediate results until a correct final result is obtained. Fig. 2 shows an example of a possible faulty solution path which can be viewed here[6] as an animation: Since (1) the lines were not shifted correctly, (2) the eigenvalues are queried, which are also identified as incorrect. Therefore, in a next step the characteristic polynomial is queried, which is entered correctly (3). Hence, the intermediate step in which an error occurred has been identified: Apparently the zeros of the characteristic polynomial were not correct. After the prompt to calculate the zeros again, they are entered correctly (4), so that associated eigenvectors must be entered. Since these are also correct (5), the adaptive solution flow guides the learner back to the original task and the lines are finally shifted correctly.

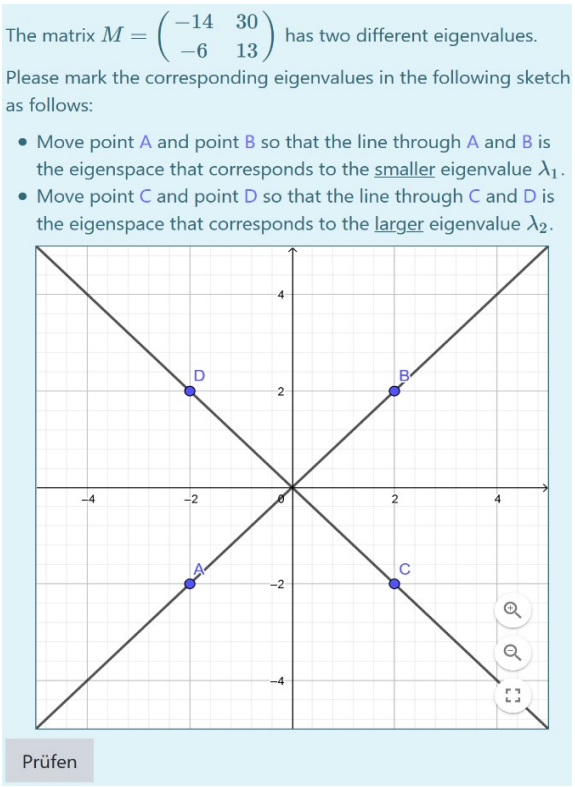

Fig. 1: STACK question combined with GeoGebra

6 https://educational-media.de/hfd-community-working-group/hfd-community-working-group/

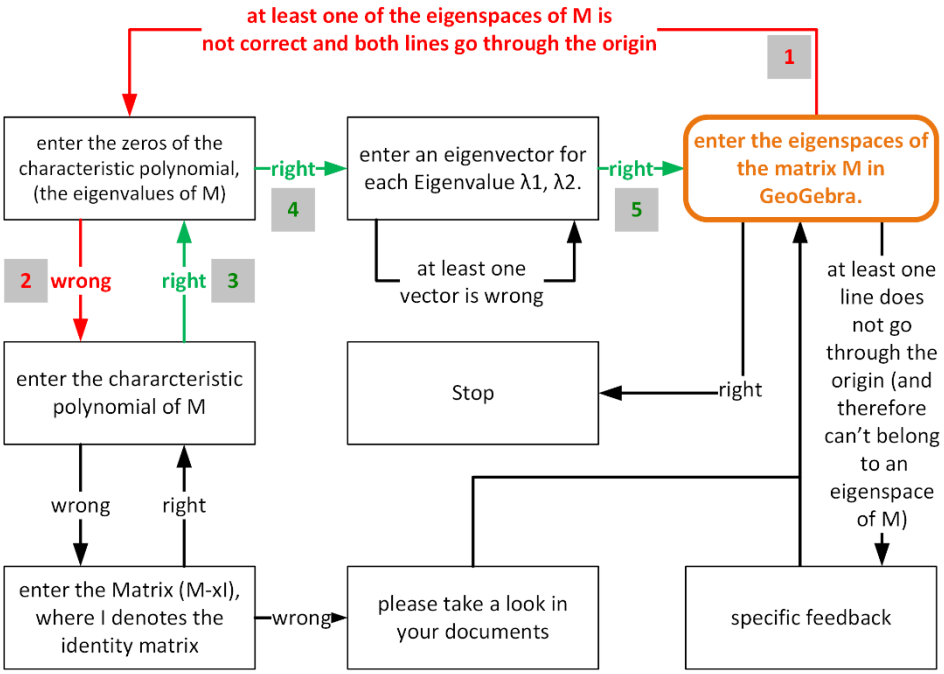

Fig. 2: Adaptive flow control within the STACK question shown in Fig. 1

3.2 Example 2: Definite Integral using Internal Adaptive Flow Control in STACK

The second example for the adaptive approach shows the calculation of a definite integral using the partial fraction decomposition.

In common implementations of the task, either a single input field is used for the end result or several input fields are implemented to query intermediate steps. Both approaches offer advantages and disadvantages. In the first approach, using only one input field for the end result, the learner is required to have complete solution competence. But, if the solution is not correct, it is not possible to identify exactly the wrong intermediate step, but only to trace possible typical errors that lead to a wrong end result. In the second approach, where input fields are used for intermediate steps, the wrong step can be identified and subsequent errors taken into account, but the learner is guided very strongly. These approaches without adaptive Flow control for the example of determination of a definite integral are visualized in Fig. .

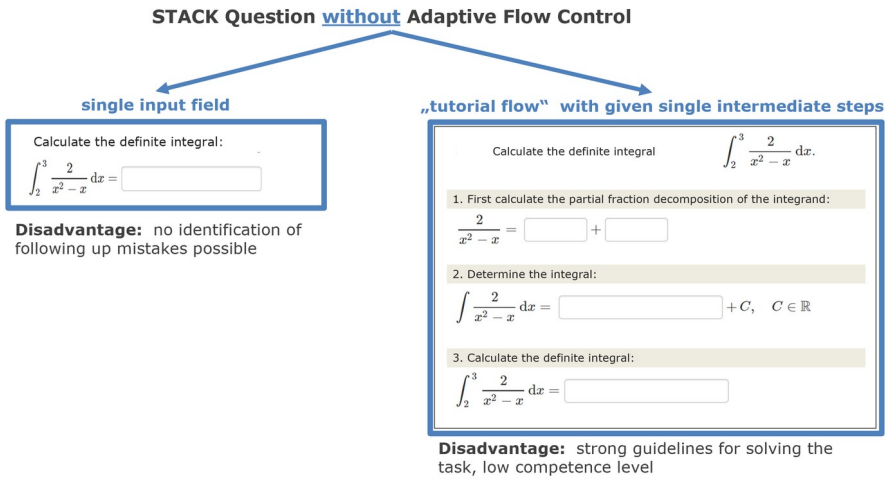

Fig. 3: STACK questions without adaptive control

The adaptive flow control combines the advantages of the mentioned two approaches: 1. the learner requires the complete solution competence and 2. it is additionally possible to identify the wrong intermediate step for calculation of the solution. If an incorrect intermediate step is detected, the learner receives an additional explanation for this step and could solve the intermediate step again. If this solution attempt is successful, the learner continues the task. The implemented adapted flow control for the calculation of a definite integral using the partial fraction decomposition is shown in Fig. 4.

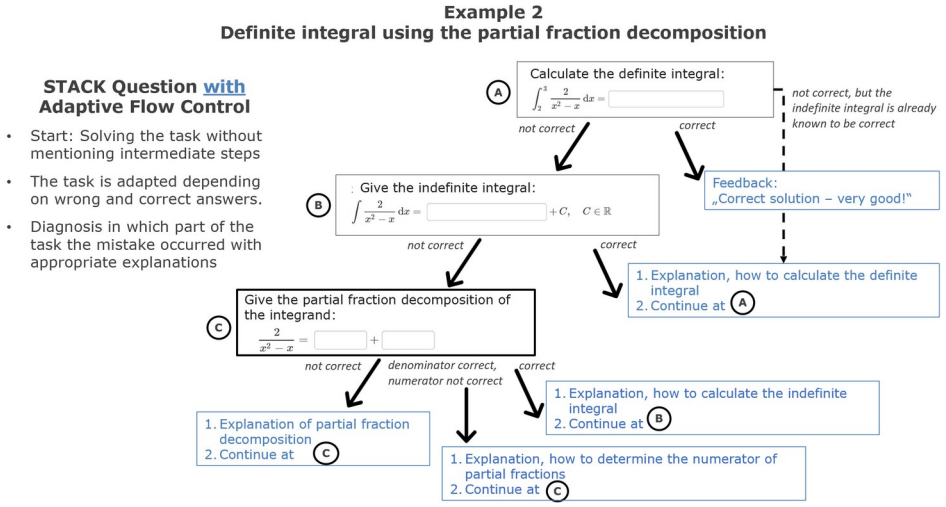

Fig. 4: Scheme of the adaptive flow control for a STACK question “calculation of a definite integral using partial fraction decomposition”

Fig. shows a specific example with a process section of internal adaptive flow control for the calculation of a definite integral using partial fraction decomposition. The figure shows some screenshots of the STACK task and the appropriate section of the scheme of the adaptive flow control of the question.

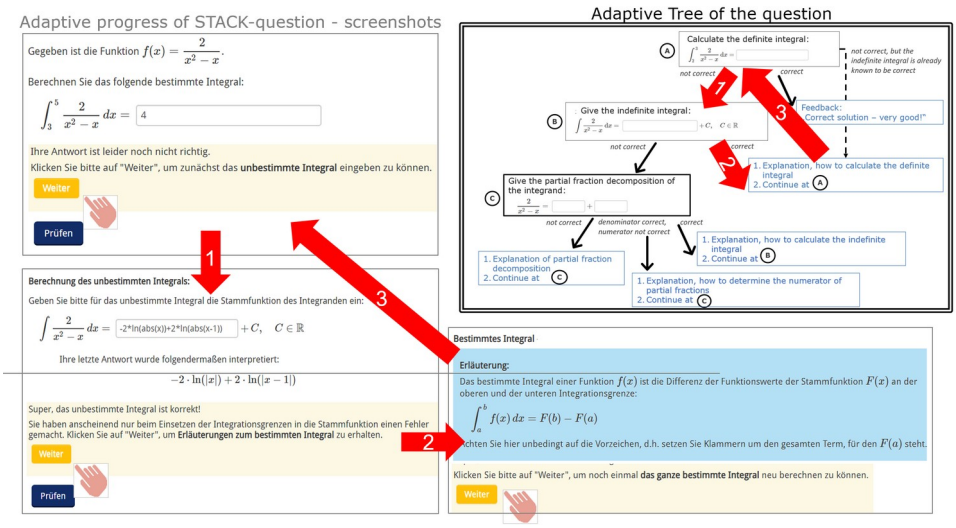

Fig. 5: Specific adapted Flow in STACK question – screenshots and scheme

4 Summary and Outlook

The first development of quizzes, exams and exercises using the new feature *internal adaptive flow control* has been done. We presented in subsection 3.1 and 3.2 two examples with adaptive flow control. The feature has already implemented in some exams and it could be observed that students like this type of tasks where they are guided to mistakes if these occur. In the next months it's planned to create more of these new exercises and to integrate them into lectures. One further next step will be to apply learning analytics to detect the most critical steps within a task.

The project is developed within a community working group of Hochschulforum Digitalisierung and supported by the German Federal Ministry of Education and Research.

The conference presentation can be viewed here: https://educational-media.de/hfd-community-working-group/hfd-community-working-group/

Bibliography

[Al19] Altieri, M.; Staupe, M.; Paluch, D.; Schirmer, E.: STACK trifft H5P – interaktives, audiovisuelles Feedback in STACK-Aufgaben basierend auf H5P. Zenodo. http://doi.org/10.5281/zenodo.2564881

[Br17] Bruder, R.; Feldt-Caesar, N.; Kallweit, M.; Krusekamp, S.; Neugebauer, C.; Schaub, M.; Winter, K.: Digital Diagnostic Testing Tasks (DDTA) - Theoretical Design and interactive example. Proceedings of the 41st Conference of the International Group for the Psychology of Mathematics Education, Volume 2, 2017.

[Ka19] Kallweit, M.; Glasmachers, E.: Adaptive Selbstlernaufgaben mit STACK. Zenodo. https://doi.org/10.5281/zenodo.2590188

[Kl19] Klischat, C.; Becker, P.; Vasko, M.: STACK is more than Maths – Development of Online-Problems for Mechanics and Electrotechnics. Zenodo. http://doi.org/10.5281/zenodo.2577116

[Sa13] Sangwin C. Computer Aided Assessment of Mathematics, Oxford University Press, ISBN: 9780199660353, 2013.

[We19] Weigel, M.; Hübl R.; Derr, K.; Podgayetskaya, T.: STACK-Aufgaben im formativen eAssessment: Einsatzmöglichkeiten des Feedbacks. Zenodo. http://doi.org/10.5281/zenodo.2563656