

Exploratives Lernen mit partiell generierter Lehr- und Lernsoftware

Andreas Kerren

Technische Universität Wien

Institut für Computergraphik und Algorithmen
Abteilung für Algorithmen und Datenstrukturen

Favoritenstraße 9-11, A-1040 Wien, Austria

Web: <http://www.ads.tuwien.ac.at/~kerren/>

Email: kerren@ads.tuwien.ac.at

Zusammenfassung: Generische bzw. generative Ansätze in der Implementierung und Anwendung von Lehr- und Lernsoftware sind bisher kaum untersucht und ebensowenig realisiert worden. Die Vorteile generischer und generativer Techniken liegen u. a. in einem hohen Grad an Wiederverwendbarkeit von Systemteilen sowie in der Reduzierung hoher Entwicklungskosten. Neben diesen Aufgaben können generative Verfahren auch zur Entwicklung bzw. Umsetzung neuer Lernmodelle benutzt werden. Der Beitrag diskutiert ein solches Lernmodell, das exploratives Lernen mit Hilfe von Generatoren propagiert. Eine Verwirklichung des Modells stellt die Lernsoftware GaniFA dar, welche die Generierung endlicher Automaten aus regulären Ausdrücken zum Gegenstand hat. Neben einer Beschreibung des Systems wird im Beitrag auch dessen Evaluation in Form von Lernexperimenten vorgestellt.

1 Einführung

Die Entwicklung interaktiver, multimedialer Lehr- und Lernsoftware (LLSW) hat während der letzten Jahre in Industrie und Ausbildung große Beachtung erfahren. Fördergelder in Millionenhöhe wurden landesweit für die Erforschung und Entwicklung neuer Medien in der Lehre bereitgestellt. *Generische* bzw. *generative* Ansätze in der Implementierung und Anwendung von LLSW sind bisher allerdings kaum erforscht, geschweige denn realisiert worden. Dabei liegen die Vorteile generischer und generativer Techniken u. a. in einem hohen Grad an Wiederverwendbarkeit von Systemteilen sowie in der Reduzierung hoher Entwicklungskosten (vgl. [Ker02, KWD00]). Beispielsweise existieren in LLSW oft die gleichen Arten von Übungen. Hier bietet sich ein guter Ansatzpunkt für den Einsatz *generativer* Werkzeuge und Verfahren, auf die wir uns in diesem Beitrag konzentrieren werden. Übungen könnten so automatisch aus Spezifikationen erzeugt werden. Es ist zu beachten, daß solche Spezifikationen nicht nur von Lehrenden angegeben werden können. Auch Lernende können die Möglichkeit der teilweisen Generierung von LLSW nutzen, wie im nachfolgenden Abschnitt 2 näher beschrieben wird.

Im Unterschied dazu wird *generische* LLSW für einen ganzen Unterrichtsbereich erstellt. Erst im Anschluß werden Instanzen für bestimmte Lerneinheiten gebildet. Ein gemeinsamer

Bestandteil von LLSW über mehrere Fächer hinweg könnte z. B. eine Lernzielkontrolle sein, wogegen andere Elemente, wie etwa Schnittstellen zu Wissensdatenbanken, eher fachspezifischen Charakter haben. Ein Beispielsystem von Diehl und Ohlmann, welches dieses generische Konzept realisiert, wird auf der Webseite [DO02] näher beschrieben.

Neben der Erzeugung von Systemteilen zur Kosteneinsparung etc. können generative Verfahren auch zur Realisation neuer Lernmodelle benutzt werden. Ein solches Lernmodell wird im Konferenzpapier [DK01]¹ diskutiert und anhand der Vermittlung von Übersetzerbauthemen und formalen Sprachen verdeutlicht. In diesem Zusammenhang werden Visualisierungen und Animationen von Berechnungsmodellen, wie zum Beispiel endliche Automaten oder abstrakte Maschinen (siehe hierzu etwa das Lehrbuch [WM96] von Wilhelm und Maurer) betrachtet, die in LLSW für den Übersetzerbau integriert werden können. Das Lernmodell läßt sich jedoch auch in anderen prozeßorientierten Disziplinen anwenden, in denen Generatoren eingesetzt werden können, so z. B. für Lerngebiete aus anderen Bereichen der Informatik, aus der Elektrotechnik oder der Physik. Es unterteilt *exploratives Lernen* [Bru61] in vier unterschiedliche Stufen. In den ersten beiden Stufen ist der Grad an Explorationsmöglichkeiten, die der Lerner wahrnehmen kann, relativ eingeschränkt: hier wird von einem statischen (Stufe 1) bzw. interaktiven (Stufe 2) Ansatz ausgegangen. Im statischen Ansatz wird die Ausführung einer Instanz eines Berechnungsmodells für eine gegebene feste Eingabe animiert. Dieser Ansatz ist stark behaviouristisch (siehe [Sch96]) geprägt, die Wissensvermittlung über die Arbeitsweise einer speziellen Instanz eines Berechnungsmodells steht im Vordergrund und der eher passive Lernende hat wenig Einfluß darauf, was ihm wie präsentiert wird. Er hat lediglich die Kontrolle darüber, wann die Animation gestartet wird oder wie schnell sie abläuft etc. Ein Beispiel für die Verwirklichung des statischen Ansatzes ist das ADLA-System von Braune et al. [BDKW99]: ein Lernsystem über die lexikalische Analyse von Programmiersprachen mittels endlicher Automaten. Im interaktiven Ansatz ist hingegen zusätzlich eine benutzerdefinierte Eingabe möglich. So kann ein Teil der Animation durch den Lerner selbst gesteuert werden, und zwar nicht nur im Hinblick auf die Kontrollierbarkeit der Animation selbst, sondern auf das, *was* zu sehen ist. Allerdings ist die visualisierte Instanz des Berechnungsmodells — wie im statischen Ansatz ebenfalls — immer noch fest vorgegeben und unveränderbar. Als Beispiel für eine LLSW, die diesem Ansatz entspricht, sei das ADSA-Lernsystem [Ker99, Ker00] zur semantischen Analyse von Programmiersprachen genannt.

Bisher sind Generatoren, die aus Spezifikationen Animationen von Instanzen eines Berechnungsmodells erzeugen können, außen vor gelassen worden. Im nächsten Abschnitt diskutieren wir allgemein, wie generative Methoden das explorative Lernen fördern können. Dazu werden die beiden höheren Stufen des oben diskutierten Lernmodells betrachtet. Anschließend wird eine generative Lernsoftware beschrieben und deren Lernerfolg anhand einer Evaluation mit mehr als 100 Versuchspersonen analysiert.

2 Lernen mit generativen Methoden

Um die grundlegende Idee von generativer LLSW besser zu verdeutlichen, zeigt Abbildung 1 eine Grobübersicht über die Struktur einer möglichen HTML-basierten LLSW, die generative Verfahren ausnutzt. Das HTML-Dokument enthält den Lernstoff über ein Themengebiet, in dem bestimmte Prozesse oder Vorgänge vermittelt werden sollen. Zusätzlich besteht die

¹Das Papier wurde vom Verfasser gemeinsam mit S. Diehl auf der ACM-Konferenz SIGCSE im Jahr 2001 veröffentlicht.

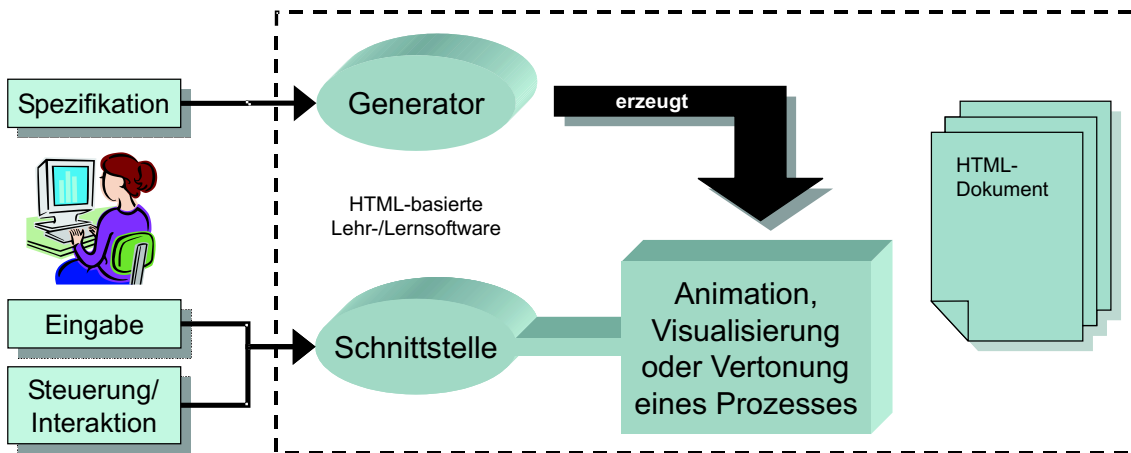


Abbildung 1: Einsatz generativer Methoden in LLSW

Möglichkeit, daß der Lerner eine Spezifikation eines im HTML-Dokument diskutierten Prozesses formuliert und diese einem Generator übergibt, der in die LLSW integriert wurde. Der Generator seinerseits erzeugt eine didaktisch möglichst sinnvolle Repräsentation des spezifizierten Prozesses, etwa eine Visualisierung (Animation) oder Vertonung. Die LLSW bietet eine — möglicherweise auch automatisch erzeugte — Schnittstelle zu dieser Repräsentation, über die der Lerner eine Eingabe für den Prozeß angibt oder die Kontrolle über den erzeugten Prozeß übernimmt, ihn steuert und mit ihm interagiert.

Der letzte Absatz diskutiert im wesentlichen den generativen Ansatz der dritten Stufe des Lernmodells aus [DK01]. Hier soll neben der reinen Wissensvermittlung verstärkt das Verstehen und die Interpretation des Lehrstoffs gefördert werden. Die Information ist nicht durch einen Lehrenden fertig strukturiert worden, sondern Lernende müssen ihre eigenen Informationen entdecken und ordnen. Dieser Ansatz (genannt „Generativer Ansatz erster Ordnung“) ermöglicht den Lernenden, neue Hypothesen zu formulieren und diese durch Änderung der Spezifikation oder der Benutzereingabe zu überprüfen. Das GANIMAM-System [DK00b] ist ein Beispiel dafür, wie der generative Ansatz erster Ordnung implementiert werden kann. Es handelt sich um einen web-basierten Generator für die interaktive Animation von abstrakten Maschinen. Der Lerner kann hier die Spezifikation einer abstrakten Maschine als Instanz eines gegebenen Maschinenmodells festlegen. Ein Generator erzeugt aus dieser Beschreibung des Lerners eine Visualisierung der spezifizierten Maschine, auf der selbstgeschriebener bzw. aus einem Hochsprachenprogramm automatisch erzeugter Maschinencode ausgeführt werden kann.

Im generativen Ansatz der vierten Stufe (genannt „Generativer Ansatz zweiter Ordnung“) können sogar Hypothesen über den Generierungsprozeß selbst überprüft werden, da der Generator in der LLSW entsprechend repräsentiert wird (z. B. über eine Animation). Dadurch wird Lernenden nicht nur der zu untersuchende Prozeß verständlich, sondern ebenfalls die Funktionsweise von Generatoren, die solche Prozesse erzeugen können. In der Lehre zum Übersetzerbau ist dies eines der zentralen Themen. Dementsprechend groß ist das Interesse daran, daß Lernende sowohl die mathematischen Grundlagen als auch die Zusammenhänge bei der Generierung von Prozessen, hier speziell von Berechnungsmodellen, verstehen. Abschnitt 3 beschreibt ein Lernsystem, das den generativen Ansatz zweiter Ordnung verwirklicht.

Die Manipulierbarkeit der Instanzen eines Prozesses über eine Spezifikationssprache führt zu einer Analogie dieses Ansatzes mit dem Mikrowelten-Konzept, welches die Konstruktion von allgemeinen Modellen propagiert, anstatt nur die Arbeit mit einem Modell zu unterstützen.

Solche Mikrowelten beschreiben nach Schulmeister [Sch96] in sich geschlossene, künstliche Umgebungen mit eigenen Regeln, die interaktiv veränderbar sind. Lernende können durch verschiedene Eingriffe auf die Umgebung einwirken oder bestimmte Bestandteile manipulieren. Mikrowelten sind somit den bekannten Simulationen, die beispielsweise in Lernsystemen für die Bereiche der Physik oder Elektrotechnik oft Verwendung finden, sehr ähnlich. Der Unterschied ist, daß sie quasi als Simulation zweiter Ordnung die Konstruktion von Modellen unterstützen, im Gegensatz zur bloßen Beschäftigung mit einem Modell selbst. Blumstengel [Blu98] führt aus, daß Mikrowelten und Simulationen oftmals aktives und exploratives Lernen fördern und bei guter Gestaltung der LLSW motivierend wirken.

Exploratives Lernen ist zudem gut mit der konstruktivistischen Sichtweise zur Gestaltung von Lernumgebungen zu vereinbaren. Im Mittelpunkt der konstruktivistischen Position steht die Auffassung, daß Wissen im Erkennensprozeß konstruiert wird. Es existiert in Relation zum Lernenden nicht autonom, sondern wird von diesem dynamisch generiert und kann deswegen nicht auf traditionelle Weise vermittelt werden. Das Wissen muß vom Lernenden selbständig in dessen bestehende mentale Struktur und Wissenskonstrukte integriert werden. Folglich wird ausgehend von einer (moderaten) konstruktivistischen Sicht verstärkt Wert auf die Schaffung und Gestaltung von stimulierenden Lernumgebungen gelegt, welche Lernenden die Möglichkeit bieten, individuelle Wissenskonstrukte zu generieren (nach [GM94, Sch96, Blu98]).

3 Fallbeispiel: Das Lernsystem GANIFA

Im Rahmen des GANIMAL-Projektes² [Gan02b] wurde eine Entwicklungsumgebung zur Realisierung des generativen Ansatzes vierter Stufe entworfen und mit ihr eine LLSW, GANIFA genannt, implementiert. GANIFA ist ein elektronisches, HTML-basiertes Textbuch, welches die Generierung endlicher Automaten aus regulären Ausdrücken zum Gegenstand hat [DKW01]. Das System kann sowohl lokal als auch über das WWW verwendet werden [Gan02a, BDK⁺02].

3.1 Systembeschreibung

Als eine Einführung in die Theorie der endlichen Automaten gibt das Textbuch eine Übersicht über formale Sprachen im allgemeinen, über reguläre Sprachen sowie über reguläre Ausdrücke. Anschließend werden Übergangsdigramme, nichtdeterministische und deterministische endliche Automaten beschrieben. Formale Definitionen werden in einem separaten Hilfsfenster des Browsers angezeigt, wenn ein Lerner auf den entsprechenden Hyperlink innerhalb der HTML-Seite klickt.

Im Übersetzerbau bekannte Techniken zur Generierung von Software wurden dazu genutzt, die in dieser LLSW enthaltenen Animationen mit Hilfe eines JAVA-Applets automatisch zu erzeugen. So ist es möglich, mit einem regulären Ausdruck beliebige endliche Automaten zu spezifizieren und den Generierungsprozeß zu animieren: Abbildung 2 zeigt zwei Bildschirmaufnahmen. Eine Seite des elektronischen Textbuchs ist in der linken Aufnahme zu sehen. Sie enthält die Theorie zur Minimierung von endlichen Automaten sowie weitere Erklärungen zu einem Algorithmus, der diese Minimierung vornimmt. Die Minimierung ist dabei als später Teil des Generierungsprozesses zu verstehen. Von dieser Seite gelangt der Lerner über einen Hyperlink zu einer weiteren Seite, auf der er einen regulären Ausdruck angeben und die Animation

²Das GANIMAL-Projekt wurde von der Deutschen Forschungsgemeinschaft (DFG) unter Aktenzeichen WI 576/8-1 und WI 576/8-3 von Juli 1998 bis Dezember 2001 gefördert.

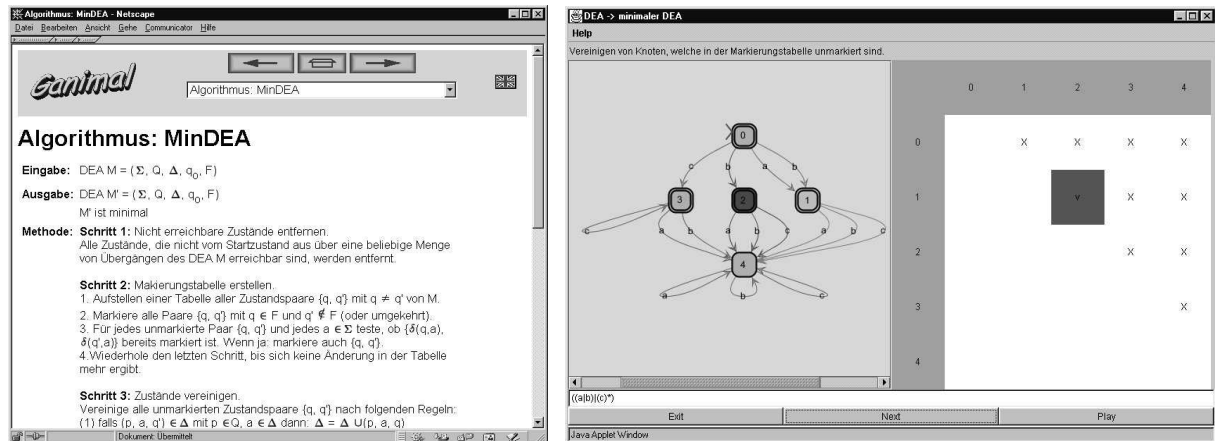


Abbildung 2: Zwei Bildschirmaufnahmen der LLSW GANIFA

des Algorithmus starten kann. Die Seite enthält zudem eine Legende über die Ereignisse des Startvorgangs und einige Zeichenerklärungen für die Animationen. Auf der rechten Aufnahme von Abbildung 2 ist die Animation eines Schritts des im Textbuch behandelten Minimierungsalgorithmus dargestellt. Der Lerner gab hier vorher den regulären Ausdruck $a|b|c^*$ an, und das Applet hat aus dieser Spezifikation zunächst einen nichtminimierten endlichen Automaten Schritt für Schritt erzeugt. Ausgehend von diesem erzeugten Automaten wird der eigentliche Minimierungsvorgang visualisiert. Jeder einzelne Generierungsschritt, vom regulären Ausdruck bis zum endlichen Automaten, kann dabei animiert werden:

1. Generierung eines nichtdeterministischen endlichen Automaten (NEA) aus einem regulären Ausdruck (RA) [WM96].
2. Beseitigung der ε -Übergänge eines NEA [RS59, WM96].
3. Transformation eines deterministischen endlichen Automaten (DEA) aus einem NEA ohne ε -Übergänge [RS59, WM96].
4. Minimierung eines deterministischen endlichen Automaten (minDEA) [HU79].
- (5.) Das GANIFA-Applet kann die Berechnung eines jeden der über die Algorithmen 1-4 erzeugten Automaten auf einem beliebigen Eingabewort visualisieren.

Die jeweilig generierten Automaten zeichnet das Applet als Übergangsdigramme. Darüber hinaus visualisiert es auch das Akzeptanzverhalten des auf diese Weise generierten Automaten für ein beliebiges benutzerdefiniertes Eingabewort, wie in Punkt 5 der obigen Aufzählung kurz erwähnt wird. Die gemachten Literaturangaben weisen auf Quellen hin, in denen die jeweiligen Algorithmen im Detail erläutert und ggf. auch deren Korrektheit bewiesen wurde. Das GANIFA-Applet kann über eine große Anzahl von Parametern an verschiedene Bedürfnisse angepaßt und auch leicht in bestehende Webseiten integriert werden. Es ist möglich, nur eine Auswahl der Algorithmen zu visualisieren und einen endlichen Automaten, regulären Ausdruck bzw. ein Eingabewort an das Applet direkt zu schicken. Zusammenfassend genügt die LLSW GANIFA aufgrund der möglichen Visualisierung des Generierungsprozesses und der automatischen Erzeugung von Animationen endlicher Automaten den Anforderungen des generativen Ansatzes zweiter Ordnung.

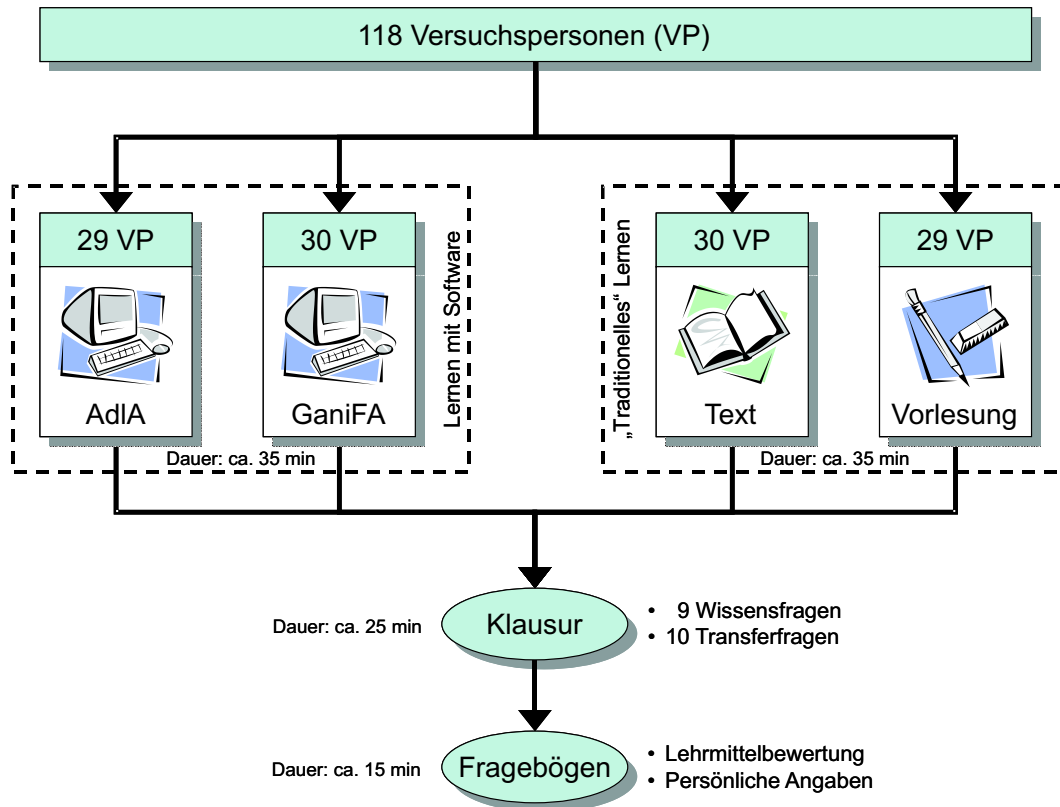


Abbildung 3: Evaluationsdesign

3.2 Evaluation

Eine Evaluation neuer LLSW soll über deren Nutzbarkeit und Effizienz Aufschluß geben. Leider werden die meisten neuen Systeme nicht ausreichend evaluiert. Der Grund mag darin liegen, daß die Möglichkeit besteht, nicht nur positives Feedback zu erhalten. Mögliche Folgen einer solchen Unterlassung sind schlechte Benutzbarkeit, Inkonsistenzen in der Darstellung oder praxisfremde Aufbereitung der Lehrinhalte. Die Akzeptanz und Leistungsfähigkeit der LLSW GANIFA wurde durch eine *summative* Evaluation (siehe z. B. [Blu98]) in Form von Lernexperimenten mit über 100 Probanden durchgeführt. Mit Hilfe eines Vergleichstestverfahrens sollte festgestellt werden, ob Personen, die mit dem generativen Lernsystem GANIFA lernten, einen vergleichbaren bzw. besseren Lernerfolg aufweisen können als Personen, die sich mit anderen Lehr- und Lernmitteln vorbereiteten.

Experimenteller Ablauf Vor der eigentlichen Evaluation wurde ein sogenannter *Usability*-Test durchgeführt, um Fehler des HTML-basierten Textbuchs aufzudecken. Tippfehler, tote Links etc. machen die Bedienung einer LLSW eher zu einer Qual und beeinflussen das Evaluationsergebnis für die LLSW schon von vorneherein negativ.

Als Versuchspersonen (VP) wurden 118 Studenten einer Informatikvorlesung des Grundstudiums ausgewählt. Die Evaluation wurde nicht angekündigt, d. h. daß die Studenten sich nicht freiwillig dazu bereit erklärt haben. Auf diese Weise wurde ausgeschlossen, daß sich nur Versuchspersonen melden, die tendenziell positiv auf interaktive multimediale LLSW eingestimmt sind oder umgekehrt. Sie wurden kurz vor Beginn der Vorlesung über das bevorstehen-

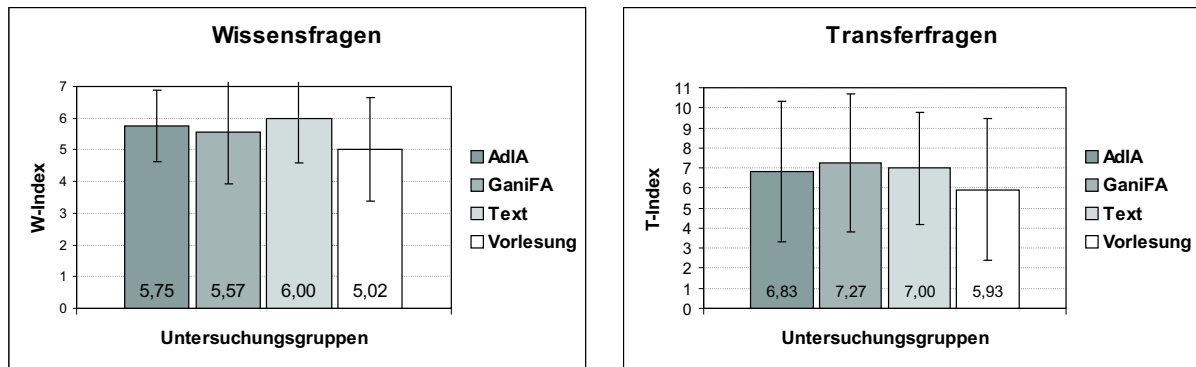


Abbildung 4: Histogramme der Mittelwerte summativer Leistungsindizes für die Wissens- (Maximalpunktzahl: 7) und Transferfragen (Maximalpunktzahl: 13) [Ker02].

de Lernexperiment informiert und zufällig in vier möglichst gleich große Gruppen aufgeteilt. Anschließend führten Mitarbeiter des GANIMAL-Projektes die Gruppenmitglieder in voneinander getrennte Räumlichkeiten. Wie in Abbildung 3 schematisch dargestellt ist, wurde ein Teilgebiet der lexikalischen Analyse in diesen vier Gruppen mit unterschiedlichen Lehr- und Lernmethoden vermittelt: Eine Gruppe lernte mit Hilfe des GANIFA-Systems, eine andere mit dem älteren, nicht-generativen ADLA-System und die beiden übrigen Gruppen konventionell mittels Lehrtext bzw. im Rahmen einer speziell vorbereiteten Vorlesung (Frontalunterricht). Die Gruppen hatten ca. 35 Minuten dafür Zeit. In jeder Gruppe wurde der identische Lehrinhalt vermittelt und Vorsorge getroffen, daß sich dessen Präsentation in den Gruppen möglichst wenig unterschied. Im Anschluß an die Lernphase wurde eine Klausur mit neun Wissens- und zehn Transferfragen gestellt. Die Wissensfragen forderten primär die Erinnerung und Reproduktion von durch das Lehrmittel explizit genannten Sachverhalten. Die Beantwortung der Transferfragen verlangte von den Versuchspersonen darüber hinaus ein tieferes Verständnis des Lerninhalts ab. Nach dem Leistungstest füllten die Probanden zusätzlich Fragebögen zur (teilweise „offen“ abzufassenden) Bewertung des jeweiligen Lehrmittels sowie zu ihrer eigenen Person aus. Mit dem Fragebogen zur Bewertung des Lehrmittels konnte gemessen werden, wie unterschiedlich gut die vier verschiedenen Präsentationsmöglichkeiten von den Probanden beurteilt wurden. Der Fragebogen zur Person half, andere Ursachen auszuschließen, die zu möglichen Effekten hätten führen können. Beide Fragebögen setzten sich aus elfstufigen Ratingskalen und numerischen Marken sowie aus sogenannten „offenen“ Fragen zusammen, zu denen die Versuchspersonen ihre Antworten schriftlich ausformulieren konnten. An dieser Stelle sei darauf hingewiesen, daß den teilnehmenden Studenten das Klausurergebnis über das WWW mitgeteilt wurde. Dazu konnten sie ihre Matrikelnummer *freiwillig* in den Klausurantworten angeben.

Ergebnisse Zunächst wurde eine deskriptive Analyse durchgeführt. Durch Aufsummierung der Punkte für die Antworten der in der Klausur gestellten Wissens- und Transferfragen ergaben sich die in Abbildung 4 gezeigten Stichprobenkennwerte. Die jeweiligen Standardabweichungen sind als Fehlerbalken repräsentiert. Für die Wissensfragen zeigt sich, daß der Leistungsmittelwert (LMW) der Textgruppe am höchsten ist ($LMW = 6,00$). Für die Transferfragen hingegen ist der Leistungsmittelwert der Gruppe am höchsten, die sich den Lernstoff mit Hilfe des elektronischen Textbuchs GANIFA angeeignet hat ($LMW = 7,27$). Dieses Ergebnis würde bei Signifikanz eines entsprechenden Tests unsere Vermutung untermauern, daß ein traditioneller

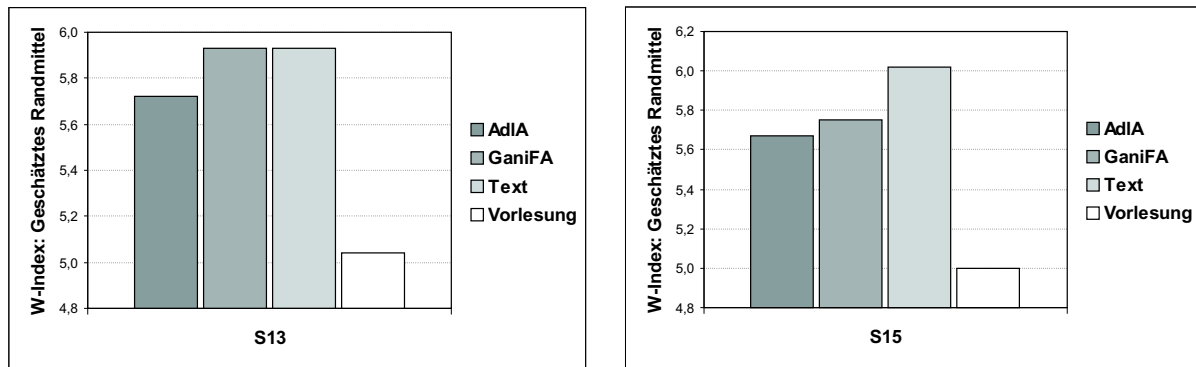


Abbildung 5: Geschätzte Leistungsdifferenzen zwischen den Lehrmethoden für die Wissensfragen unter Einbeziehung der Störvariablen S13 und S15 [Ker02].

Lehrtext besser zur Vermittlung von grundlegendem Basiswissen geeignet sein könnte als eines der beiden Lernsysteme. Wogegen die Lernsysteme ihre Vorteile in der Vertiefung und Festigung des konventionell erlernten Wissens, d. h. im Sinne einer fortgeschritteneren Stufe des Wissenserwerbs, hätten.

Eine einfaktorielle Varianzanalyse, welche die Unterschiede des Lernerfolgs zwischen den vier Gruppen auf Zufälligkeit untersucht, erbrachte allerdings weder bei den Wissens- noch bei den Transferfragen signifikante Ergebnisse. Auch eine Berechnung der Effektgröße [Bor99] brachte keine Unterschiede hervor. Erst die Einbeziehung zweier Störvariablen ergab für die Störvariable S15 „Informatikkurs in der Schule als Leistungs- bzw. Grundkurs, freiwillige Arbeitsgemeinschaft oder kein Kurs“ eine signifikante Leistungsdifferenz (Fehlerwahrscheinlichkeit: 4,1 %) bei den Wissensfragen zugunsten der Vergleichsgruppe, die mit einem äquivalenten Lehrtext gearbeitet hatte. Die GANIFA-Gruppe lag unter Kontrolle der Störvariablen S13 „Mathematik in der Schule als Leistungs- oder Grundkurs“ mit der Textgruppe gleichauf und erzielte unter Kontrolle beider Störvariablen einen Leistungszuwachs vor der ADLA- und der Vorlesungsgruppe (Fehlerwahrscheinlichkeit: 4,7 %), siehe dazu Abbildung 5. Daraus ergibt sich für das GANIFA-System und den diskutierten generativen Ansatz zweiter Ordnung eine solide Basis für weitere Entwicklungen³.

Allerdings läßt die Analyse aufgrund der sehr guten mittleren Leistungen in den Wissensfragen schließen, daß ein Teil der in unserer Evaluation gestellten Fragen möglicherweise zu einfach gewählt wurde. Effekte könnten aus diesem Grund verdeckt worden sein. Zukünftige Evaluationen sollten daher im Vorfeld des Lernexperiments einen Test durchführen, um das Vorwissen der Probanden beurteilen und angemessen berücksichtigen zu können. Da das Lernexperiment in einem zeitlich sehr stark beschränkten Rahmen stattfand, ist es zudem wahrscheinlich, daß die den Versuchspersonen zur Verfügung gestellte Zeit nicht dazu ausreichte, sämtliche Vorteile und interaktiven Möglichkeiten der LLSW vollständig aufzudecken. Hier läßt sich womöglich ein gewisser Nachteil im Evaluationsdesign feststellen, wie Baumgartner bereits im Kontext der Evaluation mediengestützten Lernens bemerkt hat [Bau97, Bau99].

Gerade die offenen Nennungen zeigten, daß sich das elektronischen Textbuch GANIFA aufgrund der Animationen, Übersichtlichkeit und besseren Motivation als Ergänzung zu üblichen Lernmethoden (etwa zu einer Vorlesung) gut eignet. Andererseits gaben einige Probanden expli-

³Eine präzisere Beschreibung der Durchführung und Analyse dieser Evaluation ist in der Dissertation des Verfassers [Ker02] enthalten.

zit an, daß sie grundsätzlich lieber aus Büchern lernen. Allgemein wurden die generierten Animationen von GANIFA als gut beurteilt. Die Versuchspersonen schätzten es sehr, eigenständig zu arbeiten und die Möglichkeiten dieser Animationen auszuschöpfen. Dies gilt ebenso für die Graphiken, die didaktische Aufbereitung und die formalen Definitionen in einem separaten Definitionsfenster. Letztere wurden allerdings als „zu formal“ angesehen. Probleme schien die nichtintuitive Eingabe der regulären Ausdrücke zu bereiten. Dies war schon vor der Evaluation abzusehen und konnte nicht mehr rechtzeitig verbessert werden. Reguläre Ausdrücke müssen in den entsprechenden Eingabefeldern der Software vollständig geklammert sein, was nicht den gängigen und somit den Versuchspersonen bekannten Konventionen entsprach. Auf Mißfallen stieß bei mehreren Versuchspersonen auch die graphische Aufmachung der HTML-Seiten und eine nicht ausreichende Erklärung der Farbsymbolik in den Animationen.

4 Zusammenfassung

In diesem Beitrag wurde ein neuer Ansatz des explorativen Lernens mit generativen Methoden präsentiert. Das als konstruktivistisch orientiert einzuordnende Lernmodell unterstützt ein selbstbestimmtes aber auch gleichermaßen selbstkontrolliertes Lernen. Eine Realisierung des Modells ist die Lehr- und Lernsoftware GANIFA, die im Kontext des DFG-Projektes GANIMAL entwickelt wurde. Nach einer Beschreibung des Lernsystems wurde eine Evaluation diskutiert, an der über 100 Versuchspersonen teilgenommen haben. Die Ergebnisse der anschließenden statistischen Analyse belegen, daß GANIFA hinsichtlich der Lerneffizienz besser als der Frontalunterricht und im Vergleich mit dem Lehrbuch nahezu gleich gut abschnitt. GANIFA wurde als Lernmethode positiv wahrgenommen, und die meisten Versuchspersonen würden GANIFA als Ergänzung zur Vorlesung gerne benutzen. Das Evaluationsdesign selbst kann jedoch kritisch betrachtet werden, da hier noch Verbesserungen hinsichtlich des Einflusses der Benutzerinteraktion im Umgang mit der Lehr- und Lernsoftware notwendig sind.

Danksagung Ich möchte S. Diehl sowie allen Mitarbeitern und Studenten, die am Gelingen des GANIMAL-Projekts beteiligt waren, für ihre tatkräftige Unterstützung danken.

Literatur

- [Bau97] P. Baumgartner. Evaluation vernetzten Lernens: 4 Thesen. In: H. Simon, Hrsg, *Virtueller Campus. Forschung und Entwicklung für neues Lehren und Lernen*, S. 131–146. Waxmann, Münster, 1997.
- [Bau99] P. Baumgartner. Evaluation mediengestützten Lernens. In: M. Kindt, Hrsg, *Projektevaluation in der Lehre – Multimedia an Hochschulen zeigt Profil(e)*, S. 61–97. Waxmann, Münster, 1999.
- [BDK⁺02] B. Braune, S. Diehl, A. Kerren, T. Weller und R. Wilhelm. *Generating Finite Automata – An Interactive Online Textbook*, 2002.
<http://www.cs.uni-sb.de/GANIMAL/GANIFA>.
- [BDKW99] B. Braune, S. Diehl, A. Kerren und R. Wilhelm. Animation of the Generation and Computation of Finite Automata for Learning Software. In: *Automata Implementation, Proceedings of the 4th International Workshop on Implementing Automata (WIA '99)*, Band 2214 der Reihe *Lecture Notes on Computer Science, LNCS*, S. 39–47, Potsdam, 1999. Springer.

- [Blu98] A. Blumstengel. *Entwicklung hypermedialer Lernsysteme*. Wissenschaftlicher Verlag Berlin, WVB, 1998.
- [Bor99] J. Bortz. *Statistik für Sozialwissenschaftler*. Springer, Berlin, Heidelberg, 5. Auflage, 1999.
- [Bru61] J. S. Bruner. The Act of Discovery. *Harvard Educational Review*, 31:21–32, 1961.
- [DK00a] S. Diehl und A. Kerren. Increasing Explorativity by Generation. In: *Proceedings of the AACE World Conference on Educational Multimedia, Hypermedia and Telecommunications (EDMEDIA '00)*, Montreal, Kanada, 2000. AACE.
- [DK00b] S. Diehl und T. Kunze. Visualizing Principles of Abstract Machines by Generating Interactive Animations. *Future Generation Computer Systems*, 16(7), 2000. Elsevier.
- [DK01] S. Diehl und A. Kerren. Levels of Exploration. In: *Proceedings of the 32nd ACM Technical Symposium on Computer Science Education (SIGCSE '01)*, S. 60–64, Charlotte, NC, USA, 2001. ACM.
- [DKW01] S. Diehl, A. Kerren und T. Weller. Visual Exploration of Generation Algorithms for Finite Automata. In: *Implementation and Application of Automata*, Band 2088 der Reihe *Lecture Notes on Computer Science, LNCS*, S. 327–328. Springer, 2001.
- [DO02] S. Diehl und M. Ohlmann. InterTalk, 2002.
<http://www.cs.uni-sb.de/~diehl/InterTalk/>.
- [Gan02a] GaniFA. Download Page, 2002.
<http://www.cs.uni-sb.de/GANIMAL/download.html>.
- [Gan02b] Ganimal. Projekt Homepage, 2002.
<http://www.cs.uni-sb.de/GANIMAL>.
- [GM94] J. Gerstenmaier und H. Mandl. Wissenserwerb unter konstruktivistischer Perspektive. Forschungsbericht Nr. 33, Ludwig-Maximilians-Universität München, März 1994.
- [HU79] J. Hopcroft und J. Ullman. *Introduction to Automata Theory, Languages and Computation*. Addison-Wesley, 1979.
- [Ker99] A. Kerren. Animation der semantischen Analyse. In: *Tagungsband zur 8. GI Fachtagung Informatik und Schule (INFOS '99)*, Informatik aktuell, S. 108–120. Springer, 1999.
- [Ker00] A. Kerren. Visualisierung und Animation der semantischen Analyse von Programmen. *Informatica Didactica – Zeitschrift für fachdidaktische Grundlagen der Informatik*, (1), 2000.
- [Ker02] A. Kerren. *Generierung interaktiver Animationen für den Übersetzerbau*. Dissertation, Universität des Saarlandes, Saarbrücken, 2002. Shaker Verlag, Aachen, ISBN 3-8322-0899-2.
- [KWD00] A. Kerren, R. Wilhelm und S. Diehl. MALL – Abschlußbericht, Juni 2000.
<http://www.cs.uni-sb.de/RW/projects/mall/>.
- [RS59] M. Rabin und D. Scott. Finite Automata and their Decision Problems. *IBM J. Res. Dev*, 3/2:115–125, 1959.
- [Sch96] R. Schulmeister. *Grundlagen hypermedialer Lernsysteme*. Addison Wesley, Bonn, 1996.
- [WM96] R. Wilhelm und D. Maurer. *Compiler Design: Theory, Construction, Generation*. Addison-Wesley, 2. Auflage, 1996.