

Integration von Algorithmenanimationen in die Lehre mittels ANIMAL

Guido Rößling

Rechnerbetriebsgruppe des Fachbereichs Informatik
TU Darmstadt
Hochschulstr. 10
64289 Darmstadt
`guido@rbg.informatik.tu-darmstadt.de`

Abstract: Algorithmenanimationen dienen zur dynamischen Präsentation von Algorithmen und Datenstrukturen. Trotz guter Forschungsergebnisse haben die Systeme noch keinen weitgehenden Einzug in die Lehre gefunden. In diesem Beitrag wird ein flexibles und gleichzeitig einfach zu nutzendes System vorgestellt. Zusätzlich wird auf dessen Einsatzmöglichkeiten in der Lehre eingegangen.

1 Einleitung

Ein wesentlicher Aspekt in der Informatik-Ausbildung ist das Verstehen dynamischer Systeme. Bereits das Grundstudium behandelt Algorithmen und Datenstrukturen, bei denen das Nachvollziehen des dynamischen Ablaufes und der zugrunde liegenden Konzepte wesentlich ist als der reine Quelltext. Hier sei etwa an „gängige“ Algorithmen wie Quicksort und Mergesort, aber auch an komplexe Datenstrukturen wie AVL-Bäume gedacht.

Zur Präsentation des Verhaltens bietet es sich an, anstelle einer statischen Präsentation (Quelltext, eventuell mit Screenshots) eine dynamische Präsentation zu nutzen, die einzelne Schritte des Verfahrens präsentieren kann. In diesem Bereich hat sich seit den späten 80er Jahren das Forschungsgebiet Algorithmenanimation etabliert, oft mit etwas anderem Fokus auch als Softwarevisualisierung oder Programmvisualisierung bezeichnet. In diesem Bereich haben sich mehrere Visualisierungssysteme etabliert, wobei zwischen generellen Systemen und themenspezifischen Ansätzen unterschieden werden kann.

Trotz aller Forschungsergebnisse und teilweise sehr ausgereifter Systeme hat es die Algorithmenanimation bislang nicht geschafft, sich fest in der Lehre zu etablieren. In diesem Beitrag wird das ANIMAL-System vorgestellt, das aufgrund seiner Flexibilität und einfachen Nutzung für mehrere Einsatzbereiche in der Lehre einsetzbar ist.

2 Das Algorithmenanimations-System ANIMAL

ANIMAL ist eines der zahlreichen aktuellen Java-basierten Systeme zur Algorithmenanimation [RF02]. Verglichen mit zahlreichen anderen Systemen wie etwa JAWAA [Ak03] bietet ANIMAL den Vorteil, dass es gleichzeitig mehrere Benutzerkreise anspricht:

- durch die grafische Schnittstelle mit WYSIWYG können Einsteiger oder Fachfremde Animationen durch Drag and Drop erstellen,
- fortgeschrittene Nutzer erstellen Animationen durch die eingebaute Skriptsprache ANIMALSCRIPT mit einem beliebigen Texteditor,
- erfahrene Programmierer können Animationen durch Nutzung einer API oder automatische Generierung von Skriptsprachen-Code erstellen.

Die Oberfläche von ANIMAL ist bewusst einfach gehalten, um möglichst viele Nutzer anzusprechen. Abbildung 1 zeigt eine Beispielanimation mit Kontrollelementen für Geschwindigkeit (links oben) und Vergrößerungsfaktor (rechts oben, hier 71%). Die Animation wird durch die Schaltflächen unten gesteuert, die Sprünge an Anfang oder Ende der Animation, Einzelschritte vor oder zurück, dynamische Anzeige eines Schrittes oder der gesamten Animation in beide Richtungen sowie Pause erlauben. Der Schieberegler rechts unten ermöglicht einen schnellen Vorlauf, normiert auf 0-100% der Gesamtanimation.

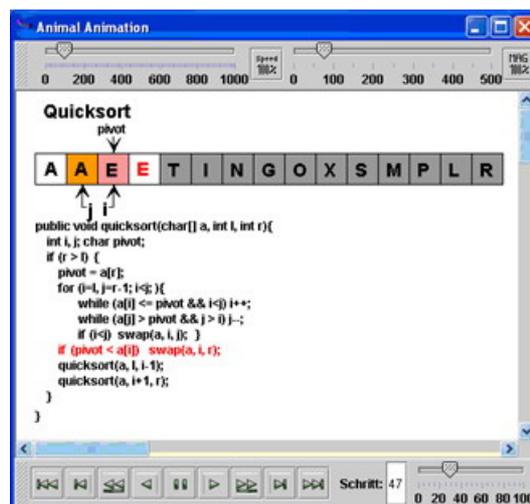


Abbildung 1: Beispiel einer Algorithmenanimation von *Quicksort*

Der Titel "Algorithmenanimation" beschreibt lediglich einen Teilaspekt von ANIMAL. Die Darstellung der Inhalte basiert ausschließlich auf der Manipulation grafischer Objekte, denen - mit wenigen Ausnahmen, wie etwa Listenelementen - keine eigene Semantik zu-

grunde liegt. Daher kann das System neben Algorithmen und Datenstrukturen auch beliebige Inhalte präsentieren. Diese Flexibilität ist auch bei Algorithmen und Datenstrukturen wichtig, da sie hier die Einbettung von erläuternden Kommentaren sowie eine textuelle oder pseudocode-basierte Beschreibung der Inhalte erlaubt.

Die Installation und Verwendung von ANIMAL ist sehr einfach und ermöglicht eine Nutzung für Dozenten und Studierende auch Informatik-ferner Fachrichtungen. Der Aufwand zur Erstellung einer Animation hängt vom gewählten Erstellungsansatz und der Vorkenntnis des Nutzers ab. Sofern der Autor bereits weiß, welche Inhalt wie präsentiert werden sollen, erfordert eine vollständige manuelle Erstellung mittels Drag and Drop in der Regel etwa 6 Stunden. Die Skriptsprache bietet einige Erweiterungen, wie etwa die Unterstützung von Listen und Felder mit den üblichen Operationen wie Setzen von Zeigern oder Vertauschen von Feldelementen [RF01]. Auch aus diesem Grund ist die Erstellung mittels ANIMALSCRIPT nach Einarbeitung in die Syntax der Sprache in der Regel signifikant schneller.

Die Nutzung von AV-Systemen wurde durch eine Arbeitsgruppe von Experten [Na03] in die folgenden sechs Hierarchiestufen eingeteilt:

1. Keine Nutzung von AV-Systemen,
2. Passives Betrachten und Steuern mittels Kontrolltasten,
3. Beantworten von Fragen zu den gezeigten Inhalten,
4. Modifizieren der Inhalte oder Eingangsdaten zum Experimentieren oder Erzielen eines bestimmten Ergebnisses,
5. Erstellen einer Visualisierung in einem AV-System,
6. Präsentation eigener oder fremder Inhalte mittels eines AV-Systems.

Die Hypothesen der Arbeitsgruppe können wie folgt zusammengefasst werden:

- Stufe 1 und 2 zeigen keinen signifikanten Unterschied im Verständnis - eine rein passive Nutzung bietet also keinen echten Mehrwert gegenüber keiner Nutzung;
- Alle Stufen ab 3 bieten jeweils einen signifikanten höheren Grad der Auseinandersetzung (Engagement) und damit auch bessere Lernchancen.

Von diesen sechs Hierarchiestufen deckt das ANIMAL-Basissystem Stufe 2 (Betrachten und Steuern) ab. In Kooperation mit dem JHAVÉ-System wird auch Bereich 3 abgedeckt [RN02]. Für Bereich 4 liegen erste Animationsgeneratoren vor, die mehrere Sortierverfahren abdecken. Der Nutzer muss hierzu die Eingabedaten vorgeben und ein Sortierverfahren wählen. Zusätzlich kann er einige Einstellungen zur grafischen Umsetzung geben, etwa Farbwahl oder Schriftgrößen. Auf Knopfdruck wird dann eine Animation gemäß den Vorgaben erstellt, die dann direkt in ANIMAL angezeigt werden kann. Die Hierarchiestufen 5 und 6 werden von Animal automatisch unterstützt. Die Erstellung eigener Inhalte wurde in mehreren Proseminaren von Studierenden ohne Probleme bewältigt, gekoppelt mit der Präsentation der dabei entstandenen Animationen.

3 Einsatz von Algorithmenanimationen in der Grundlehre

Die Integration von Algorithmenanimationen in die Grundlehre erfolgte bislang im Wesentlichen auf drei Arten:

- Das Animationssystem ist für Studierenden zum Selbststudium verfügbar, zusammen mit umfangreichen Animationen zu den behandelten Themen
- Im Proseminaren und Seminaren konzipieren Studierende eine Visualisierung eines gegebenen Sachverhalts und setzen diese in eine Animation um. Dabei wurden neben Informatikthemen auch andere Themengebiete bearbeitet.
- In Vorlesungen werden Animationen präsentiert und erläutert, vor allem in Grundzüge der Informatik I (Grundlagen, Java, Algorithmen, Grundkonzepte des Parser- und Interpreterbaus) und III (Algorithmen und Datenstrukturen).

Bei der Nutzung der Animationen in Lehrveranstaltungen erwies sich insbesondere die einstellbare Vergrößerung als hilfreich. Ohne die eigentliche Animation ändern zu müssen, konnte der Dozent für bessere Beamernutzung auf eine Vergrößerung von mehr als 100% wechseln. Die Vergrößerung ist in 1%-Schritten anpassbar, so dass der Schieberegler etwa so lange nach rechts gezogen werden kann, bis der Seitenrand minimal wird und damit die vorhandene Projektionsfläche ideal ausgenutzt wird.

Notebook-Nutzer können die Animation parallel zum Dozenten entsprechend ihrer persönlichen Lerngeschwindigkeit durchgehen. Für das Verständnis hat sich dabei insbesondere die Möglichkeit, einen Schritt in der Animation zurückzugehen, als besonders wichtig erwiesen [RN02].

Für den effektiven Einsatz in der Lehre ist es oft wichtig, gezielt zu bestimmten Abschnitten springen zu können. So sollte bei mehreren Präsentationen des gleichen Sortierverfahrens die Einführung in die Grundkonzepte übersprungen werden können. ANIMAL setzt hierzu Sprungziele ein, die vom Animationsersteller beliebig definiert werden können. Alle Sprungziele werden in einem eigenen Fenster gesammelt (Abbildung 2) und werden wie Hyperlinks durch Anklicken aktiviert.

Das Beispiel in Abbildung 2 zeigt eine (einfache) Anwendung von Quicksort, die zunächst die Grundkonzepte vorstellt. Der Lehrende kann direkt zu einzelnen Schritten der Animation springen. So können etwa die Grundkonzepte übersprungen werden oder nach einigen Schritten direkt zur Betrachtung der Komplexität gegangen werden. Die Zahl in Klammern gibt die zu dem Sprungziel gehörende Schrittzahl der Animation an.

4 Zusammenfassung und Ausblick

Das ANIMAL-System kann in mehreren Bereichen der Lehre produktiv eingesetzt werden. Aufgrund der einfachen und gleichzeitig flexiblen Nutzung eignet es sich für Lehrende und

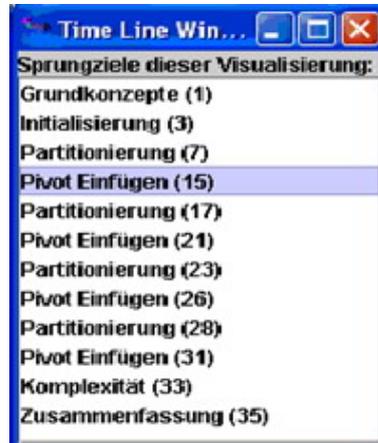


Abbildung 2: Navigation durch Sprungzielverwaltung in ANIMAL

Studierende gleichermaßen. Die in [Na03] vorgeschlagene Hierarchie zur Auseinandersetzung mit den Inhalten wird vom System weitgehend abgedeckt. Aktuelle und zukünftige Arbeiten zielen auf die kontinuierliche Erweiterung und Verbesserung der Möglichkeiten, insbesondere mit Bezug zur Interaktion. Das System steht zum freien Download unter <http://www.animal.ahrgr.de> zur Verfügung.

Literatur

- [Ak03] Akingbade, A., Finley, T., Jackson, D., Patel, P., und Rodger, S. H.: JAWAA: Easy Web-Based Animation from CS 0 to Advanced CS Courses. In: *Proceedings of the 34th ACM SIGCSE Technical Symposium on Computer Science Education (SIGCSE 2003)*, Reno, Nevada. S. 162–166. ACM Press, New York. 2003.
- [Na03] Naps, T. L., Rößling, G., Almstrum, V., Dann, W., Fleischer, R., Hundhausen, C., Korhonen, A., Malmi, L., McNally, M., Rodger, S., und Velázquez-Iturbide, J. Á.: Exploring the Role of Visualization and Engagement in Computer Science Education. *ACM SIGCSE Bulletin*. 35(2):131–152. June 2003.
- [RF01] Rößling, G. und Freisleben, B.: ANIMALSCRIPT: An Extensible Scripting Language for Algorithm Animation. *32nd ACM SIGCSE Technical Symposium on Computer Science Education (SIGCSE 2001)*, Charlotte, North Carolina. S. 70–74. February 2001.
- [RF02] Rößling, G. und Freisleben, B.: ANIMAL: A System for Supporting Multiple Roles in Algorithm Animation. *Journal of Visual Languages and Computing*. 13(2):341–354. 2002.
- [RN02] Rößling, G. und Naps, T. L.: Towards Improved Individual Support in Algorithm Visualization. *Second International Program Visualization Workshop, Århus, Denmark*. S. 125–130. June 2002.