

VISA: A Tool for Visualizing and Animating Automata and Formal Languages (Extended Abstract)

Markus Holzer¹ and Muriel Quenzer²

¹ Wilhelm-Schickard Institut für Informatik, Universität Tübingen,
Sand 13, D-72076 Tübingen, Germany
holzer@informatik.uni-tuebingen.de

² Soluz GmbH, Aspenhaustraße 5, D-72770 Reutlingen, Germany
muriel@informatik.uni-tuebingen.de

The use of multimedia tools in education has gained a lot of interest during the last decade (see, e.g., [1]). Free standing multimedia as well as tutorials distributed *via* the Internet provide the potential for students to learn on their own, at their own pace, and in their own sequence, whereas textbooks or instructors usually impose a sequence how to learn the material. Besides general reasons for using multimedia tools in education the computer provides an excellent opportunity to explain and visualize complex subjects with an abstract theoretical background.

Currently the authors develop an Internet tutorial called VISA (Visualized Automata and Formal Languages) for the WWW where hypertext together with visualized and animated standard constructions (used in proofs) provide an easy and flexible access to automata theory and formal languages [2]. The major part of the Internet tutorial is a software package, also called VISA, for displaying automata and standard constructions thereon in the WWW. VISA the software package is written in the object oriented programming language JAVATM and can also be used as a stand-alone tool to teach basics of automata theory. Due to the wide range of the material of a complete undergraduate course in automata theory and formal languages, we have restricted the first implementation phase to finite automata only.

Already existing stand-alone tools such as, e.g., AMORE, FLAP, and GRAIL, only show and explain automata and their functioning, whereas VISA is also able to visualize, animate, and illustrate mathematical constructions in a *step-by-step* manner. The idea to animate constructions (algorithmical behavior) is not new and already used in, e.g., the theory of data-structures and (computational) geometry. However, to our knowledge it has not yet been used in the above mentioned context. In the so far implemented prototype the following visualizations as JAVATM applets are available (for the constructions see, e.g., [3, 4]): (1) tracing a deterministic or nondeterministic finite automaton on a given input string, (2) step-by-step conversion of a nondeterministic finite automaton into an equivalent deterministic one (powerset construction), (3) step-by-step conversion of a finite automaton into an equivalent regular expression, and (4) step-by-step conversion of a regular expression into an equivalent finite automaton. A screenshot showing one step in the animation of (2) is given in Figure 1.

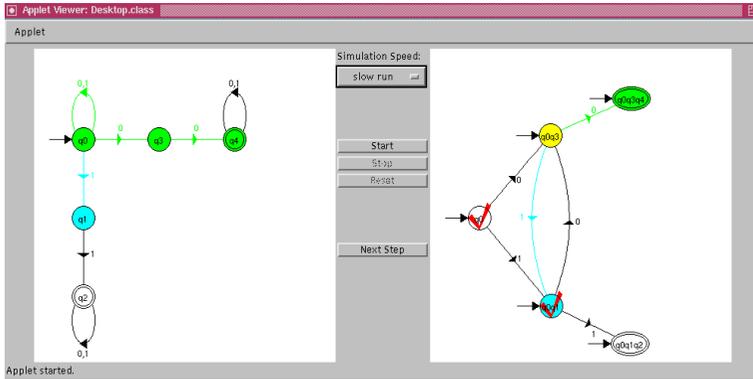


Fig. 1. One step in the animation of the powerset construction for state $\{q_0, q_3\}$ applied to the finite automaton depicted on the left-hand side. The right-hand side shows the parts of the deterministic automaton constructed so far. The development is illustrated with the help of color illumination.

Obviously, the visualization of constructions as shown in the above figure depends on good graph drawing algorithms. Since we are interested in step-by-step simulations of algorithms as well as interactive manipulations of the input objects (the automata) we need algorithms for *incremental* graph drawing. Since we have not found an appropriate incremental algorithm for our purpose, this might be a good starting point for further research.

During autumn 1998 we expect to complete a prototype of the software package and the Internet tutorial in order to use it in the undergraduate course “Einführung in die Informatik III,” which will be held at the university of Tübingen in winter semester 1998/99. In parallel, empirical studies and statistics that the package achieves its goal in real classroom applications will be done. Under URL <http://www-fs.informatik.uni-tuebingen.de/VisA/> further information on VISA will be available, soon.

References

- [1] B. Cassel and G. Davies, editors. *Integrating Technology into Computer Science*, volume 28 of *ACM SIGCSE Bulletin*. ACM, 1996.
- [2] M. Holzer and M. Quenzer. VISA: Towards a students’ green card to automata theory and formal languages. In P. Strooper, editor, *ACM Proceedings of the 3th Australasian Conference on Computer Science Education*, pages 67–75, The University of Queensland, Brisbane, Australia, 1998.
- [3] J. E. Hopcroft and J. D. Ullman. *Formal Languages and Their Relation to Automata*. Addison-Wesley, 1968.
- [4] D. Wood. *Theory of Computation*. John Wiley & Sons, 1987.