FINDING *UPPER* BOUNDS FOR NONDETERMINISTIC STATE COMPLEXITY IS HARD

Hermann Gruber and Markus Holzer
Institut für Informatik, Technische Universität München,
Boltzmannstraße 3, D-85748 Garching bei München, Germany
e-mail: {`gruberh,holzer`}`@in.tum.de`

ABSTRACT

Finite automata are one of the oldest and most intensely investigated computational models. Historically, the theory of finite automata counts among the oldest topics in computer science, and is one of the basic building blocks of the theory of formal languages. Its origins can be tracked back to the 1940's, when finite automata were used to model nerve nets [6]. Since its original discovery, finite automata have found applications both as a computing model and data structure in many application domains. Today, one of the most prominent applications of finite automata is text processing, e.g., in construction of compilers for programming languages. The power and usability of text processing tools based on regular languages like "`sed`" and "`grep`" for UNIX operating systems inspired the advent of the programming language "`perl`" with built-in capabilities for regular languages. This again prompted the decision-makers of today's leading programming platforms to build in support for regular languages. But finite automata are also found widespread use in many other different areas such as circuit design, natural language processing, computational biology, parallel processing, image compression, to mention a few [1, 7, 8, 9].

As some of these applications deal with huge masses of data, the amount of space needed by finite automata is an important research topic. On the one hand, it is well known that while nondeterministic finite automata and deterministic finite automata are equal in expressive power, nondeterministic automata can be exponentially more succinct than deterministic ones. On the other hand, minimizing deterministic finite automata can be carried out efficiently, whereas the state minimization problem for nondeterministic finite state machines is **PSPACE**-complete, even if the regular language is specified as a deterministic finite automaton [5]. This prompted the authors of the aforementioned paper to ask whether there exist at least polynomial-time approximation algorithms with a reasonable performance guarantee. However, recent work [2] shows that this problem cannot be approximated within $\frac{\sqrt{n}}{\mathrm{polylog}\, n}$ for state minimization and $\frac{n}{\mathrm{polylog}\, n}$ for transition minimization, provided some cryptographic assumption holds, namely that nonuniform logspace contains strong pseudo-random functions. As this is not a usual assumption under which inapproximability results are proved, the authors note the task of proving approximation hardness results under the weaker (and more familiar) assumption $\mathbf{P} \neq \mathbf{NP}$ as an open problem.

In this paper we solve this open problem. At first glance we show that minimizing nondeterministic finite automata, when given a deterministic finite state machine, is not approximable within $n^{1-\epsilon}$, for all $\epsilon > 0$, for an input alphabet of size $O(n)$, unless $\mathbf{P} = \mathbf{NP}$. The proof is based

on a reduction from the biclique edge cover problem. This problem turns out to be central to our studies. Note, that recently this problem was also used in [3] to obtain *lower* bounds on the nondeterministic state complexity of finite automata. With a more sophisticated construction we can overcome the problem of the growing alphabet size. In this way, we prove that minimization of nondeterministic finite automata w.r.t. the number of states with binary input alphabet is not approximable within $n^{\frac{1}{2}-\epsilon}$, for all $\epsilon > 0$, unless $\mathbf{P} = \mathbf{NP}$. When considering the number of transitions instead of the number of states, the non-approximation factor changes from $n^{\frac{1}{2}-\epsilon}$ to $n^{\frac{1}{3}-\epsilon}$. Observe, that the ratio we established for transition minimization is much lower compared to state minimization. This may be due to the fact that we used a reduction from the biclique edge cover problem on graphs, which incorporates much of our structural knowledge about nondeterministic state complexity [3, 4], but we know much less about nondeterministic transition complexity.

The noted inapproximability results naturally carry over to the setup, when specifying the input by a truth table. The approximation complexity when specifying a regular language by a truth table was stated as an open question in [2]. We show that the decision version of the problem is **NP**-complete, and that there is a polytime algorithm with performance ratio comparable to the lower bounds, approximating the problem within a factor of $O\left(\frac{\sqrt{N}\log\log N}{(\log N)^{4/3}}\right)$ in the case of state minimization, and $O\left(\frac{N\log\log N}{(\log N)^{2}}\right)$ in the case of transition minimization, where N denotes the size of the truth table. From the inapproximability point of view we prove that any efficient approximation algorithm will have an approximation factor of at least $N^{\frac{1}{4}-\epsilon}$, for all $\epsilon > 0$, unless $\mathbf{P} = \mathbf{NP}$. Again, most of the proofs rely on the biclique edge cover problem.

References

[1] J. A. Brzozowski and M. Yoeli. *Digital Networks*. Prentice-Hall, 1976.

[2] G. Gramlich and G. Schnitger. Minimizing NFA's and regular expressions. In V. Diekert and B. Durand, editors, *Proceedings of the 22nd Annual Symposium on Theoretical Aspects of Computer Science*, number 3404 in LNCS, pages 399–411, Stuttgart, Germany, February 2005. Springer.

[3] H. Gruber and M. Holzer. Finding lower bounds for nondeterministic state complexity is hard. In O. H. Ibarra and Z. Dang, editors, *Proceedings of the 10th International Conference on Developments in Language Theory*, number 4036 in LNCS, pages 363–374, Santa Barbara, California, USA, June 2006. Springer.

[4] J. Hromkovič. *Communication Complexity and Parallel Computing.* Texts in Theoretical Computer Science. Springer, 1997.

[5] T. Jiang and B. Ravikumar. Minimal NFA problems are hard. *SIAM Journal on Computing*, 22(6):1117–1141, December 1993.

[6] W. S. McCulloch and W. H. Pitts. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5:115–133, 1943.

[7] Kai Salomaa and Sheng Yu. Synchronization expressions: Characterization results and implementation. In *Jewels are Forever, Contributions on Theoretical Computer Science in Honor of Arto Salomaa*, pages 45–56. Springer, 1999.

[8] K. Sgarbas, N. Fakotakis, and G. Kokkinakis. Compact acyclic NFAs. In B. L. Webber, editor, *39th Annual Meeting on Association for Computational Linguistics*, pages 474–481. Association for Computational Linguistics, 2001.

[9] K. Čulik II and P. C. von Rosenberg. Generalized weighted finite automata based image compression. *Journal of Universal Computer Science*, 5(4):227–242, 1999.