Yale University
Department of Computer Science

The Power of Local Self-Reductions

Richard Beigel
Yale University

Howard Straubing
Boston College

YALEU/DCS/TR-1002
November 1993

The Power of Local Self-Reductions

Richard Beigel* Howard Straubing†
Yale University Boston College

Abstract

Identify a string x over $\{0,1\}$ with the positive integer whose binary representation is $1x$. We say that a self-reduction is k-local if on input x all queries belong to $\{x-1,\ldots,x-k\}$. We show that all k-locally self-reducible sets belong to PSPACE. However, the power of k-local self-reductions changes drastically between $k = 2$ and $k = 3$. Although all 2-locally self-reducible sets belong to $\mathrm{MOD}_6\mathrm{PH}$, some 3-locally self-reducible sets are PSPACE-complete. Furthermore, there exists a 6-locally self-reducible PSPACE-complete set whose self-reduction is an m-reduction (in fact, a permutation).

1. Introduction

Identify a string x over $\{0,1\}$ with the positive integer whose binary representation is $1x$. Balcazar [1] introduced lexicographical self-reductions, which on input x query only strings that are less than x. Lexicographical self-reductions are an important tool in unifying certain connections between uniform and nonuniform complexity [1]. They are also important in the study of which complexity classes may have sparse complete sets [12].

Goldsmith, Joseph, and Young [9] (independent of Balcazar) introduced near testability, and subsequently Goldsmith, Joseph, Hemachandra, and Young [8] introduced near near-testability. Both notions are special cases of lexicographical self-reductions in which the queried string (if any) is always the immediate predecessor of the input string.

The complexity of lexicographically self-reducible sets is well understood: all of them belong to EXP and some of them are $\leq_m^p$-complete for EXP [1]. The complexity

*Yale University, Dept. of Computer Science, P.O. Box 208285, New Haven, CT 06520-8285, USA. Email: beigel-richard@cs.yale.edu. Research supported in part by the National Science Foundation under grant CCR-8958528.

†Computer Science Dept., Boston College, Chestnut Hill, MA 02167, USA. Email: straubin@bcuxs1.bc.edu. Research supported in part by the National Science Foundation under grant CCR-9203208.

of near-testable sets is also well understood: all of them belong to PARITYP and some of them are $\leq_m^p$-complete for PARITYP [8]. So is the complexity of nearly near-testable sets: all of them belong to $\mathrm{PF}^{\mathrm{NP}} \circ \mathrm{PARITYP}$ and some of them are $\leq_m^p$-complete for $\mathrm{PF}^{\mathrm{NP}} \circ \mathrm{PARITYP}$ [10].

In order to better understand lexicographical self-reductions, we ask what happens when a self-reduction is allowed to look only at the k immediately preceding strings for some constant k.

Definition 1. A is k-locally self-reducible if there if a polynomial time-bounded deterministic oracle Turing machine M such that on input x

- M^A accepts if and only if x belongs to A, and
- M^A queries only elements of $\{x-1, \ldots, x-k\}$.

Remark: The 1-locally self-reducible sets are the same as the nearly near-testable sets.

We would like to say that certain self-reductions are m-reductions. Since no queries can be made on input Λ, we therefore allow m-reductions to be undefined on finitely many inputs.

Definition 2.

- A is *m-reducible* to B if there is a partial function f such that for all but finitely many x we have $x \in A \iff f(x) \in B$.
- If the partial function f above is 1–1, then A is *1-reducible* to B.
- If the partial function f above is 1–1 and onto, then A is *permutation-reducible* to B.

Our main results:

- All k-locally self-reducible sets belong to PSPACE.
- All 2-locally self-reducible sets belong to $\mathrm{MOD}_6\mathrm{PH}$. ($\mathrm{MOD}_6\mathrm{PH}$ is a generalization of the polynomial hierarchy where we allow a bounded number of MOD_6 quantifiers interspersed with the usual existential and universal quantifiers. It is an exponential analogue of ACC(6).)
- There exists a 3-locally self-reducible PSPACE-complete set.
- There exists a 6-locally self-reducible PSPACE-complete set whose self-reduction is in fact a permutation reduction. (The reader may be surprised that there is a self-reducible PSPACE-complete set whose self-reduction is even an m-reduction. The key to the reduction is determining which question to ask, rather than what to do with the answer.)

Our results are based on those of Barrington, Immerman, Straubing, and Thérien [4, 6, 5] on circuits and monoids. (But we are not the first to apply those results to Turing machine complexity classes. See [7, 11].)

2. A Connection to Algebra

Let A be k-locally self-reducible, so there is a polynomial-time algorithm $\mathcal{A}$ that takes $x+1$ and $\chi_A(x-k+1),\ldots,\chi_A(x)$ as input and determines $\chi_A(x+1)$. Then there is a polynomial-time algorithm $\mathcal{A}'$ that takes $x+1$ and $\chi_A(x-k+1),\ldots,\chi_A(x)$ as input and determines $\chi_A(x-k+2),\ldots,\chi_A(x+1)$.

We can run $\mathcal{A}'$ with input $x+1$ and each element of $\{0,1\}^k$ in succession, thus determining a finite function that maps $\chi_A(x-k+1),\ldots,\chi_A(x)$ to $\chi_A(x-k+2),\ldots,\chi_A(x+1)$. Let $\mathcal{A}^*$ be the polynomial-time algorithm that maps $x+1$ to this finite function. Then, to determine $\chi_A(x+1)$ it suffices to compute

$$(\chi_A(1),\ldots,\chi_A(k))\mathcal{A}^*(k+1)\mathcal{A}^*(k+2)\cdots\mathcal{A}^*(x+1).$$

(Here we compose finite functions from left to right.) This expression can be evaluated left to right using a constant amount of space to store the k-tuple, linear space to store the input to $\mathcal{A}^*$, and polynomial space to run $\mathcal{A}^*$. Hence we have proved our first result:

Theorem 3. *If A is k-locally self-reducible then A is in* PSPACE.

There is a more useful way to look at the preceding proof. The observant reader may have already noted that the algorithm above reduces membership in A to chain multiplication in a finite monoid consisting of certain mappings on $\{0,1\}^k$. For each k, the complexity of the chain multiplication problem depends in a simple way on the structure of the monoid. It is actually well known that the problem of multiplying n elements of a finite monoid is in NC^1 [4]; thus, the multiplication of exponentially many uniformly generated elements can be done by uniform AND–OR circuits having linear depth.

In order to prove a matching lower bound for sufficiently large k, we will use a uniform version [4] of Barrington's theorem [4] relating NC^1 to chain multiplication in S_5. Recall that S_5 is the group of all permutations on the set $\{1,2,3,4,5\}$. We apply permutations on the right. We say that a permutation π *fixes* a number i if $i\pi = i$. Let id denote the identity permutation.

Definition 4.

$$\{\mathrm{MULTS}_5 = \{\langle x_1,\ldots,x_n\rangle : x_1,\ldots,x_n \in S_5 \text{ and } x_1\cdots x_n \text{ fixes } 1\}.$$

Theorem 5 (Barrington–Immerman–Straubing [5]). MULTS_5 *is complete for* NC^1 *under dlogtime m-reductions.*

In other words, NC^1 is as easy as multiplying n elements of the symmetric group S_5 and determining whether the resulting element of S_5 fixes 1. Thus, languages recognized by uniform polynomial-depth circuits are as easy as multiplying exponentially many uniformly generated elements of S_5 and determining whether the product fixes 1.

Before we continue, it is important to point out the connection between circuits and space. Every language recognized by uniform AND–OR circuits having depth $s(n)$ is in DSPACE($s(n)$) [3, Theorem 4.2], and every language in DSPACE($s(n)$) is recognized by uniform AND–OR circuits having depth $O((s(n))^2)$ [2, Theorem 5.2]. In particular, PSPACE is the class of languages recognized by uniform polynomial-depth AND–OR circuits.

Corollary 6.

i. There is a polynomial-time computable mapping ϕ from Z^+ to S_5 such that the following language is PSPACE-complete:

$$K_\phi = \{x : \phi(x)\phi(x-1)\cdots\phi(1) \text{ fixes } 1\}.$$

ii. There is a polynomial-time computable function ψ from Z^+ to $\{id, (1,2), (2,3), (3,4), (4,5)\}$ such that the following language is PSPACE-complete:

$$K_\psi = \{x : \psi(x)\psi(x-1)\cdots\psi(1) \text{ fixes } 1\}.$$

Proof:

i. We start with any PSPACE-complete language like QBF. QBF is recognized by a polynomial-time uniform family of polynomial-depth circuits. The acceptance problem for these circuits is polynomial-time m-reducible to an exponentially long chain multiplication over S_5. Let f be that m-reduction, so $f(x)$ is an exponentially long chain of elements of S_5. By padding with the identity element if necessary, we may assume that the length of each chain depends only on the length of x. Let the infinite sequence ϕ consist of 120 copies of $f(1)$, preceded by 120 copies of $f(2)$, etc.; i.e.,

$$\phi = \cdots f(3)^{120} f(2)^{120} f(1)^{120}.$$

Because 120 is the order of S_5, $f(i)^{120}$ is the identity element for each i, so

$$f(x) = f(x)f(x-1)^{120} f(x-2)^{120} \cdots f(1)^{120}.$$

Let $\phi(i)$ be the ith rightmost element of the sequence ϕ. Because the length of the sequence $f(x)$ is easy to compute and depends only on the length of x, we can easily compute $\phi(i)$ (without needing to compute the whole sequence).

ii. Because S_5 is generated by the 2-cycles $\{(1,2),(2,3),(3,4),(4,5)\}$, we can write each element of S_5 as a product of some number of those 2-cycles. By padding with identity elements, we can write each element of S_5 as a product of a fixed number of elements of $\{id,(1,2),(2,3),(3,4),(4,5)\}$. Construct the sequence ψ by so modifying ϕ, and let $\psi(i)$ be the ith rightmost element of ψ. ∎

Theorem 7. *There exists a* PSPACE*-complete language that has a 6-local self-reduction in which the reduction is in fact a permutation.*

Proof: We define a self-reducible set S recursively. Let $\psi(\cdot)$ be as in Corollary 6.

- $1 \in S$
- $\{2,3,4,5\} \subseteq \overline{S}$
- if $q \geq 1$ and $1 \leq r \leq 5$ then, $5q + r \in S$ iff $5(q-1) + r\psi(q) \in S$

Then $x \in K_\psi$ iff $5x + 1 \in S$, so S is PSPACE-complete. Clearly, S has a 6-local self-reduction that is 1–1 and onto from $Z^+ - \{1,2,3,4,5\}$ to Z^+. ∎

Theorem 8. *There exists a* PSPACE*-complete language that has a 3-local self-reduction.*

Proof sketch: Let A be 3-locally self-reducible. Testing membership in A is equivalent to evaluating exponentially long products of certain kinds of finite functions. These functions form a finite monoid. In the appendix, we show that this particular monoid (M_3) contains the group S_5. It follows that the chain multiplication problem for M_3 is NC^1-hard under dlogtime m-reductions. As in the preceding proof, we can therefore construct a 3-locally self-reducible PSPACE-complete set. ∎

Theorem 9. *If A is 2-locally self-reducible then A belongs to* $\mathrm{MOD}_6\mathrm{PH}$ *(and, in fact, A belongs to a bounded level in* $\mathrm{MOD}_6\mathrm{PH}$ *independent of the particular 2-local self-reduction).*

Proof sketch: Let A be 2-locally self-reducible. Testing membership in A is equivalent to evaluating exponentially long products of certain kinds of finite functions. These functions form a finite monoid. In the appendix, we show that this particular monoid (M_2) contains only solvable groups. In fact, it contains only groups whose order is of the form 2^i3^j. It follows that the chain multiplication problem for M_2 is in ACC(6). Therefore we can evaluate exponentially long polynomial-uniform chain products over M_2 in $\mathrm{MOD}_6\mathrm{PH}$. (A 2-locally self-reducible set's level in $\mathrm{MOD}_6\mathrm{PH}$ is bounded by an absolute constant independent of the particular 2-local self-reduction, because the chain multiplication problem belongs to a fixed level of ACC(6).) ∎

3. Appendix: Algebra

A *semigroup* is a set equipped with an associative multiplication. A *monoid* is a semigroup with an identity element, which we denote by 1. Here we are interested in *finite* semigroups and monoids.

Let Q be a finite set. The set of maps from Q into itself forms a finite monoid with composition as the operation. Actually there are two monoids we can define this way, one in which we compose maps from right to left, the other in which we compose maps from left to right. These two monoids are not isomorphic, but they contain precisely the same groups, so for our purposes it does not really matter which one we choose. In what follows we will assume that maps are composed from left to right, so that fg means "apply first f, then g." We thus also write the image of an element of Q under a map f as qf rather than $f(q)$. We use the term *transformation monoid* to mean any monoid of maps on a finite set, in which the identity element is the identity map.

Let $f : \{0,1\}^k \to \{0,1\}$. We define

$$F_f : \{0,1\}^k \to \{0,1\}^k$$

by

$$F_f(x_1, \ldots, x_k) = (x_2, \ldots, x_{k-1}, f(x_1, \ldots, x_k)),$$

where $f : \{0,1\}^k \to \{0,1\}$ is a map. Let us denote this transformation monoid M_k. We will study the structure of M_k for various values of k.

A monoid typically contains many subsets that are closed under multiplication in the monoid, and are thus *subsemigroups* of the monoid. When the subsemigroup is a group, we call it a *group in the monoid.* (We avoid the term "subgroup" in this context, since this is often taken to mean that the identity of the group coincides with that of the monoid.)

Lemma 10 (Folklore). *Let M be a transformation monoid on a finite set Q. Every group in M is isomorphic to a permutation group on $|Q|$ elements.*

Proof: Let G be a group in M. Let e be the identity of G. If $g \in G$ then $Qg = (Qg)e$ and thus the image of g is contained in the image of e. Conversely, $Qe = (Qg^{-1})g$, and thus the image of e is contained in that of g. So all elements of G have the same image I. For any $g \in G$, $Ig = (Qe)g = Qg = I$, so every element of G permutes I. If $g, h \in G$ induce the same permutation on I, then for all $q \in Q$, $qg = (qe)g = (qe)h = qh$ and thus $g = h$. Thus G is isomorphic to a group of permutations of $I \subseteq Q$, which can be embedded in the group of permutations on $|Q|$ elements. ∎

We say that a finite monoid M is *solvable* if every group in M is solvable. There is more to this terminology than meets the eye: Every monoid M admits a certain kind of decomposition (the Krohn–Rhodes decomposition) in which the factors are simple

composition factors of the groups contained in M. When M is a solvable monoid, all of these composition factors are cyclic groups of prime order. Barrington and Thérien [6] show that in this case we can compute the product of n elements of M using an $ACC(r)$ circuit family, where r is the product of the distinct prime divisors of the cardinalities of the groups in M.

Proposition 11. *M_2 is solvable, and the prime divisors of the orders of the groups in M_2 are 2 and 3.*

Proof: By Lemma 10, every group in M_2 is isomorphic to a subgroup of S_4, the symmetric group on four letters. Thus M_2 is solvable, and the only primes that divide the order of a group in M_2 are 2 and 3. We need only show that M_2 contains both a group of order 2 and a group of order 3.

Let $f(0,1) = 0$, $f(1,0) = 1$, $f(0,0) = 0$, and $f(1,1) = 1$. Then F_f transposes $(1,0)$ and $(0,1)$ and leaves $(0,0)$ and $(1,1)$ fixed, thus generating a group of order 2 in M_2.

Let $g(0,0) = 1$, $g(0,1) = 0$, $g(1,0) = 0$, and $g(1,1) = 1$. Then F_g cycles $(0,0)$, $(0,1)$, and $(1,0)$, and fixes $(1,1)$. Thus G_g generates a group of order 3 in M_2. ∎

As a corollary, we find that multiplication of n elements in M_2 can be performed by an $ACC(6)$ family of circuits.

Proposition 12. *M_3 is not solvable.*

Proof: We define functions $f, g : \{0,1\}^3 \to \{0,1\}$ by

$$\begin{aligned} f(0,0,0) = f(0,0,1) = f(0,1,0) = f(1,1,1) = 1, \\ f(0,1,1) = f(1,0,0) = f(1,0,1) = f(1,1,0) = 0, \\ g(0,0,1) = g(0,1,0) = g(1,0,0) = g(1,1,1) = 1, \\ g(0,0,0) = g(0,1,1) = g(1,0,1) = g(1,1,0) = 0. \end{aligned}$$

Let us denote a triple $(a,b,c) \in \{0,1\}^3$ by $4a+2b+c$. With this notation, F_f and F_g are the permutations (01364)(25) and (1364)(25), respectively. In particular, F_f and F_g generate a subgroup of $S_5 \times S_2$. The projection of this group onto the left-hand component is all of S_5, because

$$(01364)(01364)(1364)(01364)(01364) = (36),$$

and any 2-cycle and 5-cycle generate S_5. Thus M_3 contains a nonsolvable group. ∎

Acknowledgments. We would like to think David Barrington and Eric Allender for helpful discussions, as well as Bill Gasarch for proofreading.

References

[1] J. L. Balcázar. Self-reducibility. *JCSS*, 41, 1990.

[2] J. L. Balcázar, J. Díaz, and J. Gabarró. *Structural Complexity I*, volume 11 of *EATCS Monographs on Theoretical Computer Science*. Springer-Verlag, New York, 1988.

[3] J. L. Balcázar, J. Díaz, and J. Gabarró. *Structural Complexity II*, volume 22 of *EATCS Monographs on Theoretical Computer Science*. Springer-Verlag, New York, 1989.

[4] D. A. Barrington. Bounded-width polynomial-size branching programs recognize exactly those languages in NC^1. *JCSS*, 38(1):150–164, Feb. 1989.

[5] D. A. M. Barrington, N. Immerman, and H. Straubing. On uniformity in NC^1. *JCSS*, 41:274–306, 1990.

[6] D. A. M. Barrington and D. Thérien. Finite monoids and the fine structure of NC^1. *J. ACM*, 35(4):941–952, Oct. 1988.

[7] J. Cai and M. Furst. Pspace survives constant-width bottlenecks. *International Journal of Foundations of Computer Science*, 2, 1991.

[8] J. Goldsmith, L. Hemachandra, D. Joseph, and P. Young. Near-testable sets. *SICOMP*, 20(3):506–523, June 1991.

[9] J. Goldsmith, D. Joseph, and P. Young. Self-reducible, p-selective, near-testable, and p-cheatable sets: The effect of internal structure on the complexity of a set. TR 87-06-02, Dept. of Computer Science, University of Washington, Seattle, June 1987. An extended abstract appeared in *Proceedings of the 2nd Annual Conference on Structure in Complexity Theory*, IEEE Computer Society Press, June 1987, pp. 50–59.

[10] L. A. Hemachandra and A. Hoene. On sets with efficient implicit membership tests. *SICOMP*, 20(6):1148–1156, Dec. 1991.

[11] U. Hertrampf, C. Lautemann, T. Schwentick, H. Vollmer, and K. W. Wagner. On the power of polynomial time bit-reductions. In *Proceedings of the 8th Annual Conference on Structure in Complexity Theory*, pages 200–207, 1993.

[12] M. Ogiwara and A. Lozano. On one query self-reducible sets. In *Proceedings of the 6th Annual Conference on Structure in Complexity Theory*, pages 139–151, 1991.