Available online at www.sciencedirect.com
SCIENCE DIRECT
ELSEVIER
Journal of Computer and System Sciences 69 (2004) 235–243
JOURNAL OF COMPUTER AND SYSTEM SCIENCES
http://www.elsevier.com/locate/jcss

Minimal DFA for testing divisibility

Boris Alexeev

510 Millstone Cir, Athens, GA 30605, USA

Received 28 September 2003; revised 21 January 2004

Abstract

We present and prove a theorem answering the question "how many states does a minimal deterministic finite automaton (DFA) recognizing the set of base-b numbers divisible by k have?"

1. Statement of the problem

The following exercise is typical in introductory texts on deterministic finite automata (DFAs): "produce an automaton that recognizes the set of binary strings that, when interpreted as binary numbers, are divisible by k." For example, exercise 1.30 in [2] asks the student to prove that the language $\{x \mid x \text{ is a binary number that is a multiple of } k\}$ is regular for each $k \geqslant 1$; explicitly presenting an automaton is the easiest solution.

The traditional (and correct) answer constructs a k-state automaton that keeps track not only of divisibility by k, but also the current residue modulo k. For example, if the input read was 1101, the machine would remember "$13 \bmod k$". The transitions between states are simple: if the automaton's current state is "$r \bmod k$", and the input symbol read is "0", it moves to state $(2r) \bmod k$; if the input symbol read is "1", it moves to state $(2r+1) \bmod k$.

(This example also generalizes to bases other than binary. Furthermore, even if the input string is encoded in base b, the canonical DFA will still have k states. It will, however, contain b transitions from each state.)

The traditional answer, unfortunately, in general fails to produce a *minimal* DFA. This paper addresses the considerably more difficult question of "how many states does a minimal DFA that recognizes the set of base-b numbers divisible by k have?" We denote this number by $f_b(k)$ and derive a closed-form expression; in the proof, we also describe the states of the minimal DFA in more detail.

The function $f_b(k)$ may be computed by algorithmic means. The author used two implementations of the Hopcroft minimization algorithm: an original Perl program and the

E-mail address: kaif@uga.edu.

0022-0000/$ - see front matter
doi:10.1016/j.jcss.2004.02.001

highly optimized AT&T FSM Package$^{\text{TM}}$. According to experts in the field, no prior work addresses the general case of this problem except through such computational alleys.

2. Interesting patterns

The function $f_b(k)$ exhibits very curious behavior. One interesting pattern considers $f_b(k)$ with b fixed and $k = x \cdot y^z$ for increasing values of z.

Example. Table of $f_b(k)$ for $b = 6$ and $k = 2^z$. (That is, $x = 1$, $y = 2$, and z ranges from 0 to 10.)

z	0	1	2	3	4	5	6	7	8	9	10
2^z	1	2	4	8	16	32	64	128	256	512	1024
$f_6(2^z)$	1	2	3	5	8	12	20	29	45	72	104
$f_6(2^{z+1}) - f_6(2^z)$	1	1	2	3	4	8	9	16	27	32	64

The successive differences of $f_6(2^z)$ are the powers of 2 and 3, sorted in increasing order!

Example. Table of $f_b(k)$ for $b = 2^2 \cdot 5 = 20$ and $k = 30 \cdot 5^z$. (That is, $x = 30$, $y = 5$, and z ranges from 0 to 6.)

z	0	1	2	3	4	5	6
$30 \cdot 5^z$	30	150	750	3750	18750	93750	468750
$f_{20}(30 \cdot 5^z)$	4	6	14	26	58	118	246
$f_{20}(30 \cdot 5^{z+1}) - f_{20}(30 \cdot 5^z)$	2	8	12	32	60	128	300

Here, the successive differences of $f_{20}(30 \cdot 5^z)$ come in increasing order from two sequences: $\{2 \cdot 4^m\} = \{2, 8, 32, 128, \dots\}$ and $\{12 \cdot 5^m\} = \{12, 60, 300, \dots\}$.

We observe that the function $f_b(k)$ manages to pick terms, in increasing order, from two unrelated sequences! At first, it is hard to imagine a formula that would produce such a function. Investigating this bizarre behavior was the starting point for this study.

3. Main result

Theorem 1. *Let* $\lambda(x, y) = \frac{x}{\gcd(x,y)}$. *Then*

$$\begin{aligned} f_b(k) &= \lambda(k, b^\infty) + \sum_{\alpha=0}^{\infty} \min\{\lambda(b^\alpha, k), \lambda(k, b^\alpha) - \lambda(k, b^{\alpha+1})\} \\ &= \min_{\mathscr{A} \geqslant 0} \left\{ \lambda(k, b^{\mathscr{A}}) + \sum_{\alpha=0}^{\mathscr{A}-1} \lambda(b^\alpha, k) \right\} \\ &= \lambda(k, b^{\mathscr{A}_0}) + \sum_{\alpha=0}^{\mathscr{A}_0-1} \lambda(b^\alpha, k), \end{aligned}$$

where $\mathscr{A}_0$ *is the smallest nonnegative integer* α *satisfying* $\lambda(k, b^\alpha) - \lambda(k, b^{\alpha+1}) < \lambda(b^\alpha, k)$.

Remark. The function $\lambda(x, y)$ is not symmetric; indeed, $\lambda(x, y) = \lambda(y, x)$ if and only if $x = y$.

We use the notation $\lambda(k, b^\infty)$ to denote $\lambda(k, b^\alpha)$ for sufficiently large α; similarly, the infinite sum can be truncated when $\lambda(k, b^\alpha) - \lambda(k, b^{\alpha+1}) = 0$. This equality certainly holds for $\alpha \geqslant \log_2 k$.

Lemma 6 shows that the three expressions in the theorem are equivalent.

To understand the expressions of $f_b(k)$ in the theorem, we may draw a table listing α, $\lambda(b^\alpha, k)$, $\lambda(k, b^\alpha)$, and $\lambda(k, b^\alpha) - \lambda(k, b^{\alpha+1})$. The first and third expressions may be understood fairly simply as written. However, the second expression is more difficult; it states that $f_b(k)$ is the minimal sum one can obtain by summing zero of more elements of the form $\lambda(b^\alpha, k)$ (as α ranges from 0 to $\mathscr{A} - 1$) and then the following value of $\lambda(k, b^\alpha)$ (that is, $\alpha = \mathscr{A}$).

Example. $b = 6$, $k = 16 = 2^4$: We can calculate $f_b(k)$ with any of the expressions above (for the third, use $\mathscr{A}_0 = 2$). The minimal terms of the first expression appear underlined below; simultaneously, the minimal "path" $8 = 1 + 3 + 4$ (in terms of the second formula above) is indicated in boldface. Note that other paths such as $15 = 1 + 3 + 9 + 2$, $9 = 1 + 8$, and $16 = 16$ (the trivial path $\mathscr{A} = 0$) yield nonminimal sums.

α	0	1	2	3	4	5	6
$\lambda(b^\alpha, k)$	$\underline{\mathbf{1}}$	$\underline{\mathbf{3}}$	9	27	81	486	2916
$\lambda(k, b^\alpha)$	16	8	**4**	2	1	1	1
$\lambda(k, b^\alpha) - \lambda(k, b^{\alpha+1})$	8	4	$\underline{2}$	$\underline{1}$	$\underline{0}$	$\underline{0}$	$\underline{0}$
$\lambda(k, b^\infty)$							$\underline{1}$

4. Corollaries to the main result

Corollary 2. *The following are upper bounds for* $f_b(k)$:

$$\begin{aligned} f_b(k) &\leqslant k && = \lambda(k, b^0) \\ f_b(k) &\leqslant 1 + \frac{k}{\gcd(k, b)} && = \lambda(b^0, k) + \lambda(k, b^1) \\ f_b(k) &\leqslant 1 + \frac{b}{\gcd(b, k)} + \frac{k}{\gcd(k, b^2)} && = \lambda(b^0, k) + \lambda(b^1, k) + \lambda(k, b^2) \end{aligned}$$

Proof. These follow immediately from the second expression in Theorem 1. $\square$

Corollary 3. *The canonical DFA described in Section* 1 *is minimal if and only if* $\gcd(k, b) = 1$ *or* $k = 2$.

Proof. The canonical DFA has k states and hence we must determine when $f_b(k) = k$.

If $\gcd(k,b)=1$ or $k=2$, the first expression of Theorem 1 immediately gives $f_b(k)=k$. Otherwise, we have $\frac{k}{\gcd(k,b)}<k-1$, and by the previous corollary,

$$f_b(k)\leqslant 1+\frac{k}{\gcd(k,b)}<k. \qquad \square$$

Corollary 4. *The successive differences of* $f_6(2^z)$ *are powers of* 2 *and* 3, *sorted in increasing order.*

Proof. Manipulation of the result of the theorem yields

$$\begin{aligned} f_6(2^z) &= \lambda(2^z,6^\infty)+\sum_{\alpha=0}^{\infty}\min\{\lambda(6^\alpha,2^z),\lambda(2^z,6^\alpha)-\lambda(2^z,6^{\alpha+1})\} \\ &= 1+\sum_{\alpha=0}^{\infty}\min\{3^\alpha\cdot\lceil 2^{\alpha-z}\rceil,\lfloor 2^{z-\alpha-1}\rfloor\} \\ &= 1+\sum_{\alpha=0}^{z-1}\min\{3^\alpha,2^{z-\alpha-1}\}. \end{aligned}$$

It is not difficult to see that as one increments $z\mapsto z+1$, a new term of the form $\min\{3^\alpha,2^{z-\alpha-1}\}$ is added, and the desired property holds. $\square$

Remark. A similar approach may be applied to the general case of $f_b(x\cdot y^z)$ for increasing values of z. In particular, we can easily prove the pattern we noticed in Section 2 for $f_{20}(30\cdot 5^z)$.

Corollary 5. *If* $b=p^n$ (p *not necessarily prime*, *but see the remark*) *and* $k=p^m\cdot x$ *with* $\gcd(x,p)=1$, *then* $f_b(k)=x+\lceil\frac{m}{n}\rceil$.

Proof. We use the first expression of the theorem:

$$\begin{aligned} f_b(k) &= \lambda(k,b^\infty)+\sum_{\alpha=0}^{\infty}\min\{\lambda(b^\alpha,k),\lambda(k,b^\alpha)-\lambda(k,b^{\alpha+1})\} \\ &= x+\sum_{\alpha=0}^{\infty}\min\{\lceil p^{n\alpha-m}\rceil,\lceil p^{m-n\alpha}\rceil\cdot x-\lceil p^{m-(n+1)\alpha}\rceil\cdot x\}. \end{aligned}$$

As long as $n\alpha<m$, $\lceil p^{n\alpha-m}\rceil=1$ and $\lceil p^{m-n\alpha}\rceil\cdot x>\lceil p^{m-(n+1)\alpha}\rceil\cdot x$. There are precisely $\lceil\frac{m}{n}\rceil$ such α (since $0\leqslant\alpha<\frac{m}{n}$), so we have

$$\begin{aligned} f_b(k) &= x+\sum_{\alpha=0}^{\lceil\frac{m}{n}\rceil-1}\{1\}+\sum_{\alpha=\lceil\frac{m}{n}\rceil}^{\infty}\{0\} \\ &= x+\left\lceil\frac{m}{n}\right\rceil, \end{aligned}$$

as desired. $\square$

Remark. If p is prime, and thus b is a prime power, this corollary completely characterizes $f_b(k)$, as *all* k can be represented in the form $p^m \cdot x$ with $\gcd(x, p) = 1$.

5. Proof of the main result

Lemma 6. *The three expressions of Theorem* 1 *are equivalent.*

Proof. By looking at the powers of a fixed prime, we see that $\lambda(b^\alpha, k)$ and $\gcd(k, b^\alpha)$ are increasing (not necessarily strictly) functions of α. It is also easy to show that $\gcd(k, b^{\alpha+1})/\gcd(k, b^\alpha)$ is decreasing, which immediately implies that $\lambda(k, b^\alpha) - \lambda(k, b^{\alpha+1})$ is decreasing. Therefore, in the sum

$$\sum_{\alpha=0}^{\infty} \min\{\lambda(b^\alpha, k), \lambda(k, b^\alpha) - \lambda(k, b^{\alpha+1})\},$$

one takes $\mathscr{A}_0$ elements from the first sequence $\{\lambda(b^\alpha, k)\}$ and then infinitely many from the second sequence $\{\lambda(k, b^\alpha) - \lambda(k, b^{\alpha+1})\}$. Telescoping the latter, one gets the other two expressions of the theorem. (The cut-off $\mathscr{A}_0$ is the smallest nonnegative integer α satisfying $\lambda(k, b^\alpha) - \lambda(k, b^{\alpha+1}) < \lambda(b^\alpha, k)$.) $\square$

Proof of Theorem 1. Constructing a DFA directly, as in Section 1, is often difficult because one must describe the transitions between states in addition to the states themselves. We will use the Myhill–Nerode Theorem and the accompanying theory of extension invariant equivalence relations to work with the states of the automaton only.

Definition. Given a language (set of strings) L over an alphabet Σ, we define the extension invariant equivalence relation $\sim_L$ associated with L as follows: strings x and y in Σ^* are equivalent ($x \sim_L y$) if for any suffix $z \in \Sigma^*$, $xz \in L$ if and only if $yz \in L$. (As is customary, Σ^* denotes the set of all finite strings over Σ. Later, we use $\Sigma^+ = \Sigma^* \backslash \{\varepsilon\}$ to denote the set of nonempty strings over Σ.)

The Myhill–Nerode Theorem [1, Theorems 3.9–10] establishes that the minimal-state automaton accepting L has, up to isomorphism, one state corresponding to each equivalence class of $\sim_L$. Therefore, the minimal-state automaton has exactly the number of states as the index of $\sim_L$. (In particular, a language L is regular if and only if $\sim_L$ has finite index.) In addition, *any* DFA recognizing L can be altered by identifying ("gluing") some states together to obtain the minimal-state automaton.

In this proof, we let Σ be the set of base-b digits and L the set of base-b numbers divisible by k. In addition, since we work with only one language at a time, we may write $x \sim y$ rather than $x \sim_L y$.

To begin, we will restate the problem equivalently in a way that will allow us to utilize modular arithmetic. Because the canonical DFA accepting L has a state for each residue modulo k, the

Myhill–Nerode Theorem implies that the minimal-state DFA will contain states that correspond to *groups of residues* modulo k. Therefore, in the pursuing analysis, rather than considering strings of digits, we discuss residues; in a way, we are projecting Σ^* onto $\mathbb{Z}_k$ (in the natural manner). For example, L now becomes very simple: instead of containing all numbers divisible by k, it contains the single residue $0 \pmod k$. To complete the reduction, we need only bother ourselves with one further

Definition. Let $r \in \mathbb{Z}_k$ be a residue modulo k and $d \in \Sigma$ a base-b digit. We define the concatenation rd to be the residue $b \cdot r + d \pmod k$. Similarly, if $d = d_{n-1} \cdots d_1 d_0 \in \Sigma^+$ is a nonempty string of digits, let the concatenation rd be what is obtained by successively concatenating individual digits:

$$rd \equiv b \cdot (b \cdot (\cdots (b \cdot r + d_{n-1}) \cdots) + d_1) + d_0) \equiv b^n \cdot r + \overline{d_{n-1} \cdots d_1 d_0} \pmod k,$$

where $\bar{d}$ denotes d interpreted as an integer. Of course, if $d = \varepsilon$, the empty string, $rd = r\varepsilon \equiv r$.

Finally, extend $\sim_L$ onto $\mathbb{Z}_k$: residues $x, y \in \mathbb{Z}_k$ are equivalent if for any string $z \in \Sigma^*$, $xz \equiv 0 \pmod k$ if and only if $yz \equiv 0 \pmod k$.

Now, suppose $\mathscr{A}$ is a nonnegative integer. We will describe

$$\lambda(k, b^{\mathscr{A}}) + \sum_{\alpha=0}^{\mathscr{A}-1} \lambda(b^\alpha, k) \tag{$*$}$$

pre-equivalence classes, each a group of residues, which will be a refinement of the equivalence classes of $\sim_L$.

The pre-equivalence classes we define naturally present themselves in *packages*, a term we borrow from computer programming to indicate collections of classes. Altogether, there are $\mathscr{A} + 1$ distinct packages, which we number $0, \ldots, \mathscr{A}$; in addition, we will sometimes refer to package $\mathscr{A}$ as the distinctive *package etcetera*. These packages come in the sizes anticipated from (*): if $0 \leqslant \alpha < \mathscr{A}$, package α contains $\lambda(b^\alpha, k)$ pre-equivalence classes, while package $\mathscr{A}$ contains $\lambda(k, b^{\mathscr{A}})$ pre-equivalence classes.

We now define the packages. Suppose $0 \leqslant \alpha < \mathscr{A}$. Package α will consist of those residues r such that there exists a string d of length α such that $rd \equiv 0$ and no smaller α works; furthermore, these residues will be grouped according to their corresponding d's. Mathematically, for each $0 \leqslant c < b^\alpha$ such that $\gcd(b^\alpha, k) \mid c$, package α contains the pre-equivalence class $\{x \mid b^\alpha \cdot x + c \equiv 0\}$, except those x that appeared in package $\alpha - 1$ or earlier. (Note that the equation $b^\alpha \cdot x + c \equiv 0$ has a solution x iff $\gcd(b^\alpha, k) \mid c$.) Because there are precisely $b^\alpha / \gcd(b^\alpha, k) = \lambda(b^\alpha, k)$ such c in the desired range, these packages have the stated sizes. Before we proceed, note that the *union* of the pre-equivalence classes in packages 0 through α consists of *all* residues x satisfying $b^\alpha \cdot x + c \equiv 0$ with $0 \leqslant c < b^\alpha$, and no others.

Package etcetera consists of the leftovers; mathematically, it is similar, but there is no restriction on c: for each $0 \leqslant c < k$ (only to avoid duplication modulo k), package $\mathscr{A}$ contains the pre-equivalence class $\{x \mid b^{\mathscr{A}} \cdot x + c \equiv 0\}$, except those x that have appeared previously. Once again, we have the necessary number of classes, since $k/\gcd(k, b^A) = \lambda(k, b^{\mathscr{A}})$.

Example. $b = 6,\ k = 16 = 2^4$: the pre-equivalence classes for $\mathscr{A} = 2$. This value of $\mathscr{A}$ was chosen so that these groups correspond to the states in the minimal DFA. Strikeouts indicate that the given value of x satisfies $b^\alpha \cdot x + c \equiv 0$ but already appeared in a previous package.

Package 0		Package 1		Package 2 (*etcetera*)	
c	**{x}**	**c**	**{x}**	**c**	**{x}**
0	{ 0 }	0	{ ~~0~~, 8 }	0	{ ~~0, 8~~, 4, 12 }
		2	{ 5, 13 }	4	{ 3, 7, 11, 15 }
		4	{ 2, 10 }	8	{ ~~2, 10~~, 6, 14 }
				12	{ ~~5, 13~~, 1, 9 }

Recall once more from the statement of the theorem that $\mathscr{A}_0$ is the smallest nonnegative integer α satisfying $\lambda(k, b^\alpha) - \lambda(k, b^{\alpha+1}) < \lambda(b^\alpha, k)$.

We make three separate claims:

1. for any $\mathscr{A}$, our pre-equivalence classes coincide with the equivalence classes of $\sim_L$ with two possible exceptions: some pre-equivalence classes may be empty and some pre-equivalence classes in package etcetera may actually be equivalent (both of these would produce an overcount);
2. for $\mathscr{A} \leqslant \mathscr{A}_0$, all the pre-equivalence classes are nonempty; and
3. for $\mathscr{A} \geqslant \mathscr{A}_0$, the classes of package etcetera are actually inequivalent.

It follows that for $\mathscr{A} = \mathscr{A}_0$, our pre-equivalence classes are precisely the Myhill–Nerode equivalence classes of $\sim_L$.

We begin by affirming (1): if two residues r and s are in the same class of package α, there exists no string d of length less than α such that $rd \equiv 0$ or $sd \equiv 0$. In addition, $r \cdot b^\alpha \equiv s \cdot b^\alpha$, so for any string d of length at least α, we have $rd \equiv sd$. Therefore, r and s are equivalent, and the pre-equivalence classes are a refinement of those of $\sim_L$.

Moreover, if r and s are in different classes and at least one of r and s is not in package etcetera, then $r \nsim s$. Indeed, if r and s are in different packages, the result is obviously true. If r and s are in different classes of the same package α with $\alpha < \mathscr{A}$, we can also conclude that $r \nsim s$ because r and s satisfy $b^\alpha \cdot x + c \equiv 0$ for different values of c; therefore, there exists a string d (namely, the d such that $\bar{d} = c$) of length α such that $rd \equiv 0$ but $sd \not\equiv 0$.

Before continuing, we note the significance of $\mathscr{A}_0$. If $\alpha \leqslant \mathscr{A}_0$, then

$$\lambda(k, b^{\alpha-1}) - \lambda(k, b^\alpha) \geqslant \lambda(b^{\alpha-1}, k) \iff k \cdot \frac{\gcd(k, b^{\alpha-1})}{\gcd(k, b^\alpha)} \leqslant k - b^{\alpha-1},$$

and if $\alpha > \mathscr{A}_0$, then

$$\lambda(k, b^{\alpha-1}) - \lambda(k, b^\alpha) < \lambda(b^{\alpha-1}, k) \iff k \cdot \frac{\gcd(k, b^{\alpha-1})}{\gcd(k, b^\alpha)} > k - b^{\alpha-1}.$$

Equipped, we proceed in order to (2). Suppose $\mathscr{A} \leqslant \mathscr{A}_0$; then, we claim that for any fixed $0 < \alpha \leqslant \mathscr{A}$ and c such that $\gcd(k, b^\alpha) \mid c$, there exists an x satisfying

$$b^\alpha \cdot x + c \equiv 0 \qquad (\dagger)$$

which does not satisfy $b^{\alpha-1} \cdot x + c' \equiv 0$ with $0 \leqslant c' < b^{\alpha-1}$. Indeed, consider all x satisfying (†) and note that these x are spaced apart equally with $\frac{k}{\gcd(k,b^\alpha)}$ separation between consecutive solutions. Multiplying these x by $b^{\alpha-1}$ yields (possibly duplicate) residues $b^{\alpha-1} \cdot x$ spaced $k \cdot \frac{\gcd(k,b^{\alpha-1})}{\gcd(k,b^\alpha)}$ apart. But, because $\alpha \leqslant \mathscr{A} \leqslant \mathscr{A}_0$,

$$k \cdot \frac{\gcd(k, b^{\alpha-1})}{\gcd(k, b^\alpha)} \leqslant k - b^{\alpha-1},$$

whence there exists an x satisfying (†) such that $(b^{\alpha-1} \cdot x) \bmod k$ is in between 1 and $k - b^{\alpha-1}$, and such an x cannot satisfy $b^{\alpha-1} \cdot x + c' \equiv 0$ with $0 \leqslant c' < b^{\alpha-1}$. Therefore, all of the classes of packages 0 through $\mathscr{A}$ are nonempty.

We finish with (3). Suppose $\mathscr{A} \geqslant \mathscr{A}_0$; it suffices to show that if $r \sim s$ and α is the minimal α such that $b^\alpha \cdot r \equiv b^\alpha \cdot s$, then $\alpha \leqslant A$. Assume the contrary: $\alpha > \mathscr{A}$. Then, r and s are both solutions of (†) for a fixed c. To derive a contradiction, we again focus on the spacing of solutions of (†). So, consider all x satisfying (†); they are spaced $\frac{k}{\gcd(k,b^\alpha)}$ apart. As before, the residues $b^{\alpha-1} \cdot x$ for x satisfying (†) are spaced $k \cdot \frac{\gcd(k,b^{\alpha-1})}{\gcd(k,b^\alpha)}$ apart. However, because $\alpha > \mathscr{A} \geqslant \mathscr{A}_0$,

$$k \cdot \frac{\gcd(k, b^{\alpha-1})}{\gcd(k, b^\alpha)} > k - b^{\alpha-1}$$

and thus there is not enough room for two distinct $(b^{\alpha-1} \cdot x) \bmod k$ in between 1 and $k - b^{\alpha-1}$. Therefore, either $b^{\alpha-1} \cdot r \equiv b^{\alpha-1} \cdot s$ or one of r and s satisfies $b^{\alpha-1} \cdot x + c' \equiv 0$ with $0 \leqslant c' < b^{\alpha-1}$. The former contradicts the minimality of α, and the second is impossible as well: without loss of generality, r satisfies such an equation. But then, there exists a string d of length $\alpha - 1$ such that $rd \equiv 0$. Because $r \sim s$, it follows that $rd \equiv sd \equiv 0$ for a string of length $\alpha - 1$, once again contradicting the minimality of α! We have reached a contradiction in all cases, therefore our assumption was false and $\alpha \leqslant \mathscr{A}$. Therefore, any two residues are "distinguished" at or before $\alpha = \mathscr{A}$, and it follows that any r and s in the package etcetera are equivalent if and only if they are in the same pre-equivalence class.

At last, we are done. □

Acknowledgments

The author would like to thank Professors Jason Eisner of Johns Hopkins University, Dana Scott and Klaus Sutner of Carnegie Mellon University, and Michael Sipser of MIT for answering queries about the problem and related issues and Professor Rodney Canfield of the University of Georgia for many useful conversations and attention to this work.

For large-scale computations, when speed was crucial, AT&T Research's FSM Package was used to compute $f_b(k)$, to complement the author's own programs.

References

[1] J. Hopcroft, J. Ullman, Introduction to Automata Theory, Languages, and Computation, Addison-Wesley, Reading, MA, 1979.
[2] M. Sipser, Introduction to the Theory of Computation, PWS Publishing, 1997.