

CDM: Notes on Divisibility and State Complexity

KLAUS SUTNER

June 17, 2004

1 Divisibility and Recognizability

A *linear recurrent numeration system* \mathcal{U} consists of a strictly increasing sequence (U_n) of positive integers given by a linear recurrence (of order r)

$$U_n = c_1 U_{n-1} + \dots + c_r U_{n-r} \quad (1)$$

with non-negative integer coefficients c_i and initial conditions $1 = U_0 < U_1 < \dots < U_r$. The associated digit alphabet has the form $\Sigma_D = \{0, 1, \dots, D-1\}$ where D is the ceiling of the supremum of U_{n+1}/U_n . We can determine D by computing the Frobenius-Perron eigenvalue of the companion matrix. The value of a string in Σ_D^* in standard base \mathcal{U} is given by

$$\nu(x_k x_{k-2} \dots x_1 x_0) = \sum_{i \leq k} x_i U_i.$$

We allow leading 0's so the representation is not unique. More generally, we do not impose any additional constraints on the representations of numbers. Thus, while the standard Fibonacci representation of $11 = F_2 + F_3 + F_4 + F_5$ is 10100 we will also admit 1111. A set X of natural numbers is *recognizable in base \mathcal{U}* or *\mathcal{U} -recognizable* if there is a finite state machine that accepts those words over the digit alphabet Σ_D that denote numbers in X . Note that the most significant digit is supposed to appear first in a string. Alternatively, we can consider the first letter to be the least significant digit. The associated value function is

$$\nu_R(x_0 x_1 \dots x_k) = \sum_{i \leq k} x_i U_i$$

so that $\nu_R(x) = \nu(x^R)$ where x^R denotes the reversal of string x . We will refer to this numeration system as *reverse base \mathcal{U}* , denoted \mathcal{U}^R . While recognizability in base \mathcal{U} is equivalent with recognizability in reverse base \mathcal{U} the state complexity of the associated automata will differ in general.

The most important special case of a linear recurrent numeration system is of course the classical base B representation where $U_n = B^n$ for some integer $B \geq 2$. For emphasis we will refer to these systems as *radix B* and *reverse radix B* .

It is easy to see that arithmetic progressions of the form $m_0 + m\mathbb{N}$ are recognizable in any base \mathcal{U} . However, determining the state complexity of the associated minimal automata turns out to be rather difficult. We will here focus on automata for $m\mathbb{N}$, the set of multiples of a fixed multiplier m .

See [5] and [2] for background.

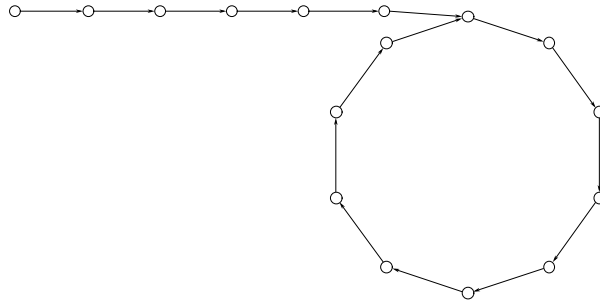
2 Canonical Recognizers

2.1 Preliminaries

Fix some modulus m and a linear numeration system \mathcal{U} . Write $t = \text{trn}(\mathcal{U}, m)$ and $p = \text{per}(\mathcal{U}, m)$ for the transient and period of the sequence $u_n = U_n \bmod m$. We refer to $\vec{u} = (u_0, u_1, \dots, u_t, \dots, u_{t+p-1})$ as the *fundamental sequence* of \mathcal{U} modulo m . In order to be able to iterate through this sequence, define the generalized remainder function by

$$\text{rem}_{t,p}(i) = \begin{cases} i & \text{if } i < t, \\ t + (i - t) \bmod p & \text{otherwise.} \end{cases}$$

Thus for any position $0 \leq i < t + p$ in the sequence, $\text{rem}_{t,p}(i + 1)$ describes the position of the next element. Of course, when $t = 0$ this agrees with the ordinary remainder function. Define a successor operation by $\text{succ}(q) = \text{rem}_{t,p}(q + 1)$ and write $\mathbb{N}_{t,p}$ for the semimodule one-generated by succ , i.e. the structure $\langle \{0, 1, \dots, t + p - 1\}, \text{succ} \rangle$.



Likewise we can define a partial, multi-valued predecessor function pred by $\text{pred}(i) = j$ iff $\text{succ}(j) = i$. When the transient is 0 the predecessor function is simply the inverse of the successor function.

Note that even for simple numeration systems it is rather difficult to determine transients and periods with respect to all moduli. As a case in point, consider the Fibonacci system $\langle 1, 1; 1, 2 \rangle$. Clearly, the transients are 0 in this case, but the periods exhibit rather complicated behavior. These numbers are sometimes referred to as Pisano numbers, see sequence A001175 in Sloane's catalog of integer sequences, [6] and the references there.

Example 2.1 Let $u_n = B^n \bmod m$ be the orbit of 1 under the map $x \mapsto Bx \bmod m$ where $m, B \geq 2$. Define $m_0 = \text{gcd}(m, B^m)$ and $m_1 = m/m_0$. Then the transient length is $\min(i \geq 0 \mid B^i = 0 \pmod{m})$ which is the maximum of all $\lceil \nu_p(m)/\nu_p(B) \rceil$ where p is any prime dividing both m_0 . The period is $\text{ord}(B; \mathbb{Z}_{m_1})$, the multiplicative order of B modulo m_1 .

2.2 Standard Base \mathcal{U}

Consider a linear numeration system \mathcal{U} together with a fixed modulus m . Let t and p be the transient and period of the fundamental sequence (u_n) . To construct a canonical DFA that tests divisibility it is easiest to first consider a nearly deterministic automaton \mathcal{A}' : since we do not know the length of the input string ahead of time, there is no way to determine the appropriate multiplier u_q for the most significant digit. Thus the automaton has state set $Q = \mathbb{Z}_m \times \mathbb{N}_{m,p}$ and the transitions are given by

$$(p, q) \cdot a = (p + a u_q \bmod m, \text{pred}(q)).$$

Initial states are $I = \{(0, q) \mid q < t\}$, and the unique final state is $(0, 0)$.

Proposition 2.1 *In \mathcal{A}' we have $I \cdot w = \{(\nu(w0^q), q) \mid 0 \leq q < t\}$.*

Proof. If \mathcal{U} is periodic modulo m the transitions function of \mathcal{A}' is complete and deterministic, so that \mathcal{A}' is a t -entry automaton in this case. If the transient is non-zero observe that precisely the states $(p, 0)$ are incomplete whereas the states (p, t) are nondeterministic of degree 2 for each letter of the alphabet. It follows that $|I \cdot w| = t$ for any word w . Indeed, $I \cdot w = \{(p_0, 0), (p_1, 1), \dots, (p_{t-1}, t-1)\}$. \square

Now let $\mathcal{A} = \mathcal{A}_{m,\mathcal{U}}$ be the accessible part of the Rabin-Scott power automaton of \mathcal{A}' . By the proposition we can identify the state set of \mathcal{A} with sequences in \mathbb{Z}_m of length T . Note that all these sequences satisfy the recurrence of \mathcal{U} modulo m so that we can truncate after the first r terms where r is the order of the recurrence. Let $\vec{u} = (u_0, \dots, u_{r-1})$. The action of \mathcal{A} on a state $\vec{p} = (p_0, \dots, p_{r-1})$ is given by

$$\vec{p} \cdot a = \sigma(\vec{p}) + a\vec{u} \bmod m$$

where $\sigma(\vec{p}) = (p_1, \dots, p_{r-1}, p_r)$ denotes the shift operation to the next r -block in the sequence.

Proposition 2.2 *The canonical DFA $\mathcal{A}_{m,\mathcal{U}}$ for divisibility testing in linear base \mathcal{U} is transitive. If \mathcal{U} is periodic modulo m then $\mathcal{A}_{m,\mathcal{U}}$ is also codeterministic. The state complexity of $\mathcal{A}_{m,\mathcal{U}}$ is at most m^d .*

In the periodic case $\vec{p} \cdot 0$ is essentially just a cyclic rotation so that the size of the state set of \mathcal{A} is not affected by truncating the alphabet to $\Sigma_2 = \{0, 1\}$. In general, though, the whole alphabet Σ_D is required. For example, the recurrence $\langle 3, 1; 1, 2 \rangle$ has dominant eigenvalue $(1 + \sqrt{13})/2 \approx 2.30$. Modulus $m = 9$ produces a transient of 2 and a period of 3. The alphabet Σ_2 generates 36 states whereas the full alphabet Σ_3 produces 81 states.

2.3 Reverse Base \mathcal{U}

Reverse base \mathcal{U} is somewhat easier to deal with since we can generate the terms (u_n) as we scan the input. The canonical DFA $\mathcal{A}^R = \mathcal{A}_{m,\mathcal{U}}^R$ again has state set $Q = \mathbb{Z}_m \times (T)$ where $T = \text{trn}(m, \mathcal{U}) + \text{per}(m, \mathcal{U})$. The transitions are given by the right action

$$(p, q) \cdot a = (p + a u_q \bmod m, \text{succ}(q)).$$

The unique initial state is $(0, 0)$ and the final states are of the form $(0, q)$ for $q < t$.

Note that \mathcal{A}^R in general fails to be transitive. For example, \mathcal{A}^R may contain a trap state.

Given a DFA \mathcal{A} for standard base \mathcal{U} an alternative approach to building a recognizer for reverse base \mathcal{U} is to apply the usual Rabin-Scott power automaton construction to the reversal of \mathcal{A} . As it turns out, this power automaton is already minimal. To see why, define a state p in a finite state machine M with state set Q to be *rich* if $\llbracket p \rrbracket_M - \llbracket Q - \{p\} \rrbracket_M \neq \emptyset$. A machine is rich if all of its states are.

Proposition 2.3 *For any DFA M the reversal automaton $\text{rev}(M)$ is rich.*

Proof. Let p be any state in $\text{rev}(M)$. Then $\llbracket \text{rev}(M) \rrbracket_{\mathcal{A}q} = \llbracket M_p \rrbracket_{\mathcal{A}q_0}^R$ where M_p is the DFA obtained from M by selecting p as its sole final state and q_0 is the initial state of M . Since M is deterministic our claim follows. \square

The last argument actually shows that the behaviors in the reverse automaton are disjoint, but for our purposes richness suffices; see [7] for other applications of this method. Since $\text{rev}(\mathcal{A})$ is rich the standard Rabin-Scott power set construction will produce the minimal DFA,

Corollary 2.1 *The power automaton $\text{pow}(\text{rev}(\mathcal{A}))$ is minimal.*

Proof. Suppose Q is the state set of the nondeterministic machine $\text{rev}(\mathcal{A})$ and let $P_1 \neq P_2 \subseteq Q$ be two states in the power automaton $\mathcal{A}' = \text{pow}(\text{rev}(\mathcal{A}))$. Without loss of generality assume $p \in P_1 - P_2$. Since p is rich in $\text{rev}(\mathcal{A})$ there is some word w that is not in the $\text{rev}(\mathcal{A})$ -behavior of any state other than p , and in particular not in the $\text{rev}(\mathcal{A})$ -behavior of P_2 . But then the \mathcal{A}' -behavior of P_1 differs from the \mathcal{A}' -behavior of P_2 and we are done. \square

As was noted by Brzozowski, one can therefore construct the minimal automaton by repeating this step: $\text{pow}(\text{rev}(\text{pow}(\text{rev}(\mathcal{A}))))$ is the minimal DFA, for any automaton \mathcal{A} . Surprisingly, very often this construction seems to provide better performance than the determinization followed by minimization, even if Hopcroft's algorithm is used for the latter operation.

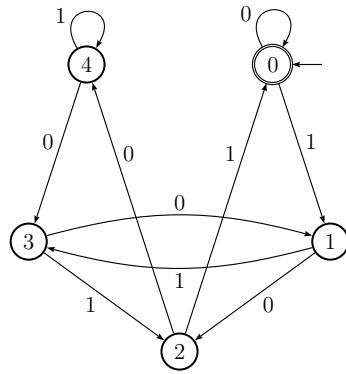
2.4 Horner Automata

In the special case of the traditional radix B numeration system the evaluation of a digit string comes down to evaluating a polynomial at B . This fact gives rise to a simple construction of a canonical Horner DFA $\mathcal{H} = \mathcal{H}_{m,B}$ for divisibility testing in radix B . \mathcal{H} has state set \mathbb{Z}_m and the right action corresponds to evaluation of a polynomial using Horner's method:

$$p \cdot a = Bp + a \bmod m.$$

0 is the initial and sole final state.

Example 2.2 Here is the diagram of the Horner automaton for $m = 5$, $B = 2$.



This particular automaton turns out to be minimal. However, in general the canonical Horner automata fail to be minimal. The table below shows the sizes of the minimal DFAs for $2 \leq m, B \leq 16$.

	B													
m	2	2	2	2	2	2	2	2	2	2	2	2	2	2
3	2	3	3	2	3	3	2	3	3	2	3	3	2	3
3	4	2	4	3	4	2	4	3	4	2	4	3	4	2
5	5	5	2	5	5	5	5	2	5	5	5	5	2	5
4	3	4	6	2	6	4	3	4	6	2	6	4	3	4
7	7	7	7	7	2	7	7	7	7	7	7	2	7	7
4	8	3	8	5	8	2	8	5	8	3	8	5	8	2
9	3	9	9	4	9	9	2	9	9	4	9	9	4	9
6	10	6	3	6	10	6	10	2	10	6	10	6	3	6
11	11	11	11	11	11	11	11	11	2	11	11	11	11	11
5	5	4	12	3	12	4	5	7	12	2	12	7	5	4
13	13	13	13	13	13	13	13	13	13	2	13	13	13	13
8	14	8	14	8	3	8	14	8	14	8	14	2	14	8
15	6	15	4	6	15	15	6	4	15	6	15	15	2	15
5	16	3	16	8	16	3	16	9	16	5	16	9	16	2

It follows from section 2.3 that we can construct a DFA for reverse radix B whose states are precisely those sets of modular numbers that are co-reachable in \mathcal{H} . But these are precisely all the solution sets of modular equations $B^kx + c = 0 \pmod{m}$ where $0 \leq c < B^k$.

3 State Complexity and Radix Bases

The canonical Horner DFAs $\mathcal{A}_{m,B}$ for radix B constructed in the last section fail to be minimal in general. They do provide upper bounds for the state complexity of $\mathcal{L}_{m,B}$, though, and are helpful in constructing the minimal automata. Recall that any DFA for a regular language covers the corresponding minimal DFA, so we need to determine the fibers of the covering map; or, equivalently, the partition of the canonical state set induced by the covering map.

Let $\mathbb{N}_{k,B} = \{c \in \mathbb{N} \mid 0 \leq c < B^k\}$. It is well-known that for radix B representation the regularity of $\mathcal{L}_{m,B}$ stems from the fact that the equivalence relation on \mathbb{N} defined by

$$x \equiv_{m,B} y \iff \forall k \geq 0, c \in \mathbb{N}_{k,B} (B^k x + c \in X \leftrightarrow B^k y + c \in X), \quad (2)$$

see [4, 3], trivially has finite index when $X = m\mathbb{N}$. The index is none other than the state complexity of $\mathcal{L}_{m,B}$.

Note that we are not interested in the automaticity of the sequence $(n \bmod m)_n$ here; since the sequence assumes m values any automaton determining the remainder from the radix B representation would by necessity have at least m states.

3.1 Solution Sets of Linear Modular Equations

In order to determine the index of $\equiv_{m,B}$ fix a base $B \geq 2$ and a modulus $m \geq 2$. For $k \geq 0$ consider the two-parameter equation

$$B^k x + c = 0 \pmod{m}. \quad (3)$$

We will refer to k as the *level* of the equation. Coefficient c is said to be *feasible* if $\gcd(B^k, m) \mid c$ and *admissible* if in addition $0 \leq c < B^k$. We write S_c^k for the set of solutions of equation (3). We will tacitly assume that coefficients are always feasible so that all solution sets are non-empty. The empty solution set which arises when m and B fail to be coprime will be dealt with separately. Define the *depth* of m and B to be the largest level k for which some S_c^k is different from all the S_d^l for $l < k$. Define the *strict solution set* for parameters k and c to be $T_c^k = S_c^k - \bigcup_{l < k, d} S_d^l$. The *strict depth* of m and B is the largest level k for which some strict solution set is non-empty. The structure of the unrestricted solution sets is easy to describe.

Proposition 3.1 *Let c be a feasible coefficient. Then*

- $|S_c^k| = \gcd(B^k, m)$.
- $S_0^k = \{x \in \mathbb{Z}_m \mid x = 0 \pmod{m/\gcd(B^k, m)}\}$.
- $S_c^k = S_0^k + d \pmod{m}$ for some suitable offset d .

To lighten notation, we write g_k for $\gcd(B^k, m)$. The *weak quotient* of x and y is defined to be $x \setminus y = x / \gcd(x, y)$. Alternatively, $x \setminus y = \text{lcm}(x, y) / y$. Let $\gamma_k = B^k \setminus m$ and $\bar{\gamma}_k = m \setminus B^k$. Let σ be the least k such that $\gcd(B^k, m) = \gcd(B^{k+1}, m)$, $\kappa = \lfloor \log_B m \rfloor$ and $\kappa^+ = \lceil \log_B m \rceil$. Thus the size of S_c^k is g_k and the elements in this set are spaced at distance $\bar{\gamma}_k$. Note that the sequences (g_k) and $(\bar{\gamma}_k)$ become stationary at σ .

Proposition 3.2 *The depth of m and B is $\max(\sigma, \kappa^+)$. The strict depth of m and B is κ^+ .*

Let s_k be the number of solution sets at level k and likewise t_k the number of strict solution sets other than \emptyset . The following lemmata describe the number of (strict) solution sets up to the depth of m and B .

Lemma 3.1 *The number of solution sets for each level k is given by*

$$s_k = \begin{cases} \min(\gamma_k, \bar{\gamma}_k) & \text{if } k \leq \sigma, \\ \gamma_k - \gamma_{k-1} & \text{if } \sigma < k < \kappa^+, \\ \bar{\gamma}_k - \gamma_{k-1} & \text{if } \sigma < k = \kappa^+, \end{cases}$$

Proof. It follows from 3.1 that no two solution sets at distinct levels up to σ can be the same. At these levels, γ_k is the number of admissible coefficients for $k \leq \kappa$, and $\bar{\gamma}_k$ otherwise. Hence, in the case where $\kappa < \sigma$ our claim follows since the last two cases do not apply.

In the case where $\kappa \geq \sigma$ the argument for the first case is similar, so suppose $\sigma < k < \kappa^+$. The new admissible coefficients at level k are the feasible coefficients c for which $B^{k-1} \leq c < B^k$. The third case is entirely similar. \square

Lemma 3.2 *The number of strict solution sets for each level k is given as follows. For $\kappa < \sigma$ we have*

$$s_k = \begin{cases} \gamma_k & \text{if } k \leq \kappa, \\ \min(\bar{\gamma}_{\kappa^+}, \bar{\gamma}_\kappa - \gamma_\kappa) & \text{if } k = \kappa^+. \end{cases}$$

Otherwise

$$s_k = \begin{cases} \gamma_k & \text{if } k \leq \sigma, \\ \bar{\gamma}_k - \gamma_{k-1} & \text{if } \sigma < k \leq \kappa, \\ \bar{\gamma}_{\kappa^+} - \gamma_{\kappa^+} & \text{if } \sigma < k = \kappa^+. \end{cases}$$

Proof. First assume that $\kappa < \sigma$, whence $\kappa^+ = \kappa + 1 \leq \sigma$. For $k \leq \kappa$ the claim follows as in the last lemma. For $k = \kappa^+$ let $q = g_{\kappa^+}/g_\kappa > 1$. In order for a strict solution set at level κ^+ to be non-empty the corresponding solution set must contain an element that, at level κ , appears at some feasible but not admissible coefficient. Divide the feasible coefficients at level κ into q blocks so that the last block has size $\bar{\gamma}_{\kappa^+}$. Since a solution set at level κ^+ is the union of q solution sets at level κ , one from each block, s_{κ^+} is none other than the number of inadmissible coefficients in this block. Our claim follows.

It remains to deal with the case $\kappa \geq \sigma$. The argument here is very similar to the last lemma and will be omitted. \square

3.2 Standard Radix B

Let $\mathcal{B} = \mathcal{B}_{m,B}$ be the minimal DFA obtained from the canonical Horner automaton $\mathcal{H} = \mathcal{H}_{m,B}$ as in section 2.4. First note that the behavior of a state p in \mathcal{H} is

$$\llbracket p \rrbracket = \{ w \in \Sigma_D^* \mid B^{|w|}p + \nu(w) = 0 \pmod{m} \} \quad (4)$$

and this behavior is non-empty for any p . Define the *witness* for p to be the length-lex minimal word in its behavior. Observe that two states in \mathcal{A} are equivalent if and only if they have the same witness. Since the length of the witness matters, rather than just its numerical value $\nu(w)$,

behavioral equivalence is thus closely connected to the strict solution sets of equation (3): the behavioral equivalence classes of \mathcal{A} are none other than the strict solution sets of (3).

The following result is established in [1].

Theorem 3.1 *The state complexity of the minimal DFA recognizing multiples of m in radix B , is*

$$\begin{aligned} \text{sc}(m, B) &= \bar{\gamma}_\alpha + \sum_{0 \leq i < \alpha} \gamma_i \\ &= \bar{\gamma}_\sigma + \sum_{0 \leq i} \min(\gamma_i, \bar{\gamma}_i - \bar{\gamma}_{i+1}) \end{aligned}$$

where α is minimal such that $\bar{\gamma}_\alpha - \bar{\gamma}_{\alpha+1} < \gamma_\alpha$.

Proof. It is not hard to show that for $\kappa < \sigma$ we have $\alpha = \kappa$ for $\bar{\gamma}_\kappa - \bar{\gamma}_{\kappa+1} < \gamma_\kappa$ and $\alpha = \kappa^+$ otherwise. For $\kappa \geq \sigma$ we have $\alpha = \sigma$. The equality of both summations is shown in the reference, so the claim follows from lemma 3.2. \square

As is shown in the reference, it follows that the canonical Horner automaton $\mathcal{H}_{m,B}$ is minimal if, and only if, $m = 2$ or m and B are coprime. The states in the Horner automaton whose witnesses have length k are those that appear in strict solution sets at level k . Thus there are t_k such states in \mathcal{B} for $k \leq \kappa^+$, the strict depth of m and B .

3.3 Reverse Radix B

Since the canonical DFA for reverse radix B is slightly more complicated than the one obtained from reversing the Horner automaton for standard radix B we use the latter to determine the state complexity of the minimal DFA.

Theorem 3.2 *Let $\tau = \min(\kappa, \sigma)$ and set $n = \sum_{k \leq \tau} \gamma_k + \sum_{\tau < k \leq \sigma} \bar{\gamma}_k$. Then the size of the minimal DFA recognizing multiples of m in reverse radix B is n whenever m and B are coprime, and $n + 1$ otherwise.*

Proof. We first show that size of the minimal DFA recognizing multiples of m in reverse radix B is the number of solution sets of equation (3). Since the power automaton obtained from $\text{rev}(\mathcal{H})$ is minimal by corollary 2.1 it suffices to count the number of states in the full power automaton that are accessible from the initial state $\{0\}$. By the same argument as in proposition 2.3 these sets are precisely the solution sets of equation (3).

Recall that the depth of m and B is $\tau = \max(\sigma, \kappa^+)$. It is easy to check that $\sum_{k \leq \tau} s_k = n$ where the terms s_k are given by lemma 3.1. Hence, when m and B are coprime n is the total number of solution sets. Otherwise we have to add 1 to account for the empty set. \square

4 Base \mathcal{U}

4.1 The Periodic Case

Clearly (U_n) is periodic modulo m if and only if the last coefficient c_r and m are coprime.

Let $T = \text{per}(m, \mathcal{U})$ be the period of (u_n) modulo m . Recall from section 2 that the canonical automaton \mathcal{A}' has state set $Q = \mathbb{Z}_m \times \mathbb{Z}_T$ and the transitions are given by

$$(p, q) \cdot a = (p + a u_{q'} \bmod m, \text{succ}(q)).$$

Lemma 4.1 *Let \mathcal{A}' be the canonical n -entry automaton that checks divisibility in base \mathcal{U} . Then the power automaton $\text{pow}(\mathcal{A})$ is reduced.*

Proof. Since \mathcal{A}' has a unique final state the reverse automaton $\text{rev}(\mathcal{A}')$ is a DFA. By proposition 2.3 it follows that the power automaton of \mathcal{A}' must already be minimal. \square

One might suspect that the Pisano number of m plays a role in the size of the minimal divisibility testing DFA \mathcal{C} for numbers in Fibonacci base. Surprisingly, this turns out not to be the case. To construct a canonical divisibility DFA \mathcal{C} set $T = \text{per}(m)$ and let $F = (F_0, F_1, \dots, F_{T-1}) \bmod m$ be one period of the Fibonacci sequence modulo m . The state set $Q \subseteq \mathbb{Z}_m^T$ of the canonical DFA is obtained by the action

$$P \cdot a = \text{rot}(P) + aF \bmod m.$$

Here rot denotes cyclic rotation to the left. The initial state is $(0, 0, \dots, 0)$ and all sequences $(p_0, p_1, 0, p_3, \dots, p_{T-1})$ are final. The only a priori bound on the size of the automaton obtained by this construction is m^n , but it turns out that its size is actually m^2 . To see this note that the states of \mathcal{C} are all Fibonacci type sequences modulo m but with different initial conditions. All initial conditions occur since the standard sequence F has the form $(0, 1, \dots, 1)$.

To show that the bound m^2 is tight, write 2 for the input string $0^{n-1}1$ so that $P \cdot 2 = P + F \pmod{m}$. Letting $P = (p_0, p_1, 0, \dots, p_{T-1})$ we have $P \cdot 0^{i-1}2^j0^{T-1}$ is final if and only if $p_i + j = 0 \pmod{m}$. Hence all states in \mathcal{C} have distinct behavior and the automaton is minimal.

Theorem 4.1 *The size of the minimal DFA recognizing multiples of m in Fibonacci base is m^2 .*

4.2 Weak Periods

Let $\vec{a} = (a_i)$ be a sequence in \mathbb{Z}_m such that $a_0 = 1$. For a positive integer p and $c \in \mathbb{Z}_m$ define (p, c) to be a *weak period* of \vec{a} if for all $i \geq 0$: $a_{i+p} = c \cdot a_i$. Informally, we also refer to p as the weak period, and c as its associated *multiplier*. Since $a_0 = 1$ these multipliers must all lie in the multiplicative subgroup \mathbb{Z}_m^* . Note that for $c = 1$ we obtain the standard notion of period.

Lemma 4.2 *Let p and q be two weak periods of \vec{a} . Then $\text{gcd}(p, q)$ is also a weak period of \vec{a} .*

Proof.

We may assume without loss of generality that p and q are coprime, otherwise we can consider blocks of length $\gcd(p, q)$. Suppose $p < q$, say, $q = ep + r$ where $0 < r < p$. Note that r and p must also be coprime. Let c and d be the associated multipliers.

By comparing terms in the sequence we find

$$\begin{aligned} da_i &= c^e a_{i+r} & \text{for } 0 \leq i < p - r \\ da_i &= c^{e+1} a_{i+r} & \text{for } p - r \leq i < p \end{aligned}$$

Repeated application of these equations shows that $a_{i+1} = da_i$. More precisely, let $0 < s < p$ be the inverse of $-r$ in \mathbb{Z}_m^* and set $s' = \lceil sr/p \rceil$. Letting $c' = (c^e d^{-1})^s c^{s'}$ it is easy to show that $(1, c')$ is a weak period of \vec{a} . \square

It follows from the last lemma that there is a uniquely determined minimal weak period π .

Theorem 4.2 *Let π be the minimal weak period of \vec{u} modulo m . The state complexity of the minimal DFA recognizing divisibility by m in reverse base \mathcal{U} is $m\pi$.*

Proof. Consider the canonical DFA \mathcal{A}^R on state set $Q = \mathbb{Z}_m \times \mathbb{Z}_n$ where n is the period of \vec{u} . Behavioral equivalence of two states (p, q) and (p', q') in \mathcal{A}^R means that for any word x we have

$$p + \nu_R^q(x) \equiv p' + \nu_R^{q'}(x)$$

where $\nu_R^q(x) = \sum x_i u_{q+i}$ and $\alpha \equiv \beta$ if either $\alpha = 0 = \beta$ or $\alpha \neq 0 \neq \beta$. Note that \equiv is not a congruence. We may safely assume that $0 \leq q \leq q'$, so that $D = q' - q \geq 0$ is the shift factor between the two summations.

Claim: (p, q) and (p', q') are behaviorally equivalent if, and only if, $p' = cp$ for some $c \in \mathbb{Z}_m^*$ and $(p' - p, c)$ is a weak period.

To see this, first assume that $p' = cp$ for some weak period $(p' - p, c)$. Then $p' + \nu_R^{q'}(x) = c(p + \nu_R^q(x))$ and $(p, q), (p', q')$ are equivalent, as required. For the opposite direction choose a word x that selects the term $u_0 = 1$ in the summation for $\nu_R^q(x)$ exactly $m - p$ times. Thus $p + \nu_R^q(x) = 0 \pmod{m}$ and it follows that $p' + \nu_R^{q'}(x) = p' + (m - p)u_D = p' - pu_D = 0 \pmod{m}$. Hence $p' = pu_D \pmod{m}$. We write $c = u_D$. Similarly, for any position s , we can choose a word x that selects the term u_s once in the sum, and the term $u_0 = 1$ r times where r is minimal such that $p + u_s + r = 0 \pmod{m}$. But then $p' + u_{s+D} + r = 0 \pmod{m}$ and $p' + u_{s+D} + rc = 0 \pmod{m}$ and therefore $cu_s = u_{s+D} \pmod{m}$. Since $u_{nD} = 1 \pmod{m}$ we must have $c \in \mathbb{Z}_m^*$, and we are done with the claim.

It follows from the claim that behavioral equivalence is determined by the minimal weak period π and the associated multiplier. \square

4.3 The Nonperiodic Case

MISSING

References

- [1] B. Alexeev. Minimal DFAs for divisibility testing. <http://arXiv.org/abs/cs/0309052>, 2003.
- [2] J.-P. Allouche and J. Shallit. *Automatic Sequences*. Cambridge UP, 2003.
- [3] V. Bruyère and G. Hansel. Recognizable sets of numbers in nonstandard bases. *LNCS*, 911:167–197, 1995.
- [4] V. Bruyère, G. Hansel, C. Michaux, and R. Villmaire. Logic and p -recognizable sets of integers. *Bull. Belg. Math. Soc.*, 1:191–238, 1994.
- [5] D. Perrin. Finite automata. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, pages 2–57. Elsevier, 1990.
- [6] N. J. A. Sloane. The on-line encyclopedia of integer sequences. www.research.att.com/~njas/sequences.
- [7] K. Sutner. Reduced power automata and sofic systems. *Foundations of Computer Science*, 14(6):1117–1128, 2003.