DRAFT: Deterministic Periodic Countable Automata

Brendan Lucier

Version 1.0, 2005/01/02

Abstract

We define a new automaton, the Deterministic Periodic Countable Automaton (DPCA). We prove properties of the languages expressable by DPCAs (which we call Countably Regular languages). In particular, we give examples of languages that are (and are not) countably regular, explore closure properties of countably regular languages, and develop a variant of regular expressions that express the countably regular languages.

1 Introduction

2 Automata Definitions

2.1 Deterministic Periodic Countable Automata

A Deterministic Periodic Countable Automata (DPCA) is a deterministic automaton $M = (Q, \Sigma, \delta, q_0, F)$ such that

1. $Q = \mathbb{Z}$,
2. $q_0 = 0$,
3. $F$ is a periodic subset of $\mathbb{Z}$, and
4. there exists a constant $C$ with $|\delta(q, a) - q| \leq C$ for all $q, a$.

Since $Q$ and $q_0$ are fixed, we shall denote a DPCA as $M = (\Sigma, \delta, F)$. Item 3 can be rewritten as “there exists a constant $T > 0$ such that $q \in F \iff q + T \in F$.” We shall denote a DPCA with particular constants $C$ and $T$ as a DPCA($C, T$).

2.2 Counted Deterministic Finite Automata

A Counted Deterministic Finite Automaton (CDFA) is simply a DFA with an integer counter. This counter starts at zero and can be incremented or decremented on each transition. The transition function depends on the value of the counter; it might move to different states depending on the counter value (not just check if the counter is zero).

Formally, a C DFA is a deterministic automaton $M = (Q, \Sigma, \delta, q_0, F)$ where the machine state is given by $(q, n) \in Q \times \mathbb{Z}$. The starting state is $(q_0, 0)$. The transition function is $\delta: Q \times \mathbb{Z} \times \Sigma \rightarrow Q \times \mathbb{Z}$ with the property that if $\delta(q, n, a) = (q', n')$ then $|n' - n| \leq 1$. That is, the integer counter can only be incremented or decremented by 1 (or left the same) on each step.
2.3 CDFA Transition Diagrams

The purpose of this section is to describe the format of CDFA transition diagrams. A CDFA diagram is drawn similarly to a DFA diagram: circles represent states, double-circles represent final states, and labelled arrows represent transitions. However, CDFA diagrams require extra information in transition labels.

A CDFA transition is labelled as $c/P/\alpha$, where $c$ is a character in the alphabet, $P$ is a subset of $\mathbb{Z}$, and $\alpha$ is a symbol chosen from $\{+, -, 0\}$. This transition is interpreted as “Take this transition if a $c$ is read and the value of the counter is contained in $P$. If this transition is taken, modify the counter according to $\alpha$.

Note that a CDFA in which all transitions are of the form $c/\mathbb{Z}/0$ is equivalent to a DFA, since the counter is not used. Note also that for each state $q$ in the CDFA, and each $c$ in the alphabet, all $P_i$ such that $c/P_i/\alpha$ is a transition from $q$ must be distinct. In a complete CDFA, the union of those sets $P_i$ will also be precisely $\mathbb{Z}$ for each character $c$. Otherwise, we assume the presence of an unescapable error state just as in DFA diagrams.

For an example of a CDFA diagram, see Figure 1.

3 Machine Equivalences

**Theorem 3.1.** DPCAs and CDFAbs are equivalent in expressing languages.

**Proof.** Choose some DPCA $(C, T) = (\Sigma, \delta, F)$. We wish to construct CDFA $M'$ such that $L(M') = L(M)$.

Choose some $k \geq 1$ such that $C \leq kT$. Since $F$ is $T$-periodic, $F$ will also be $kT$-periodic. Let $Q' = \{0, 1, 2, \ldots, kT - 1\}$ and $F' = F \cap Q'$. Then there is a natural isomorphism $\phi: Q' \times \mathbb{Z} \rightarrow \mathbb{Z}$ given by $\phi(q, n) = q + nkT$. The inverse of $\phi$ is given by $\phi^{-1}(n) = (n \mod kT, [n/kT])$. Note also that $\phi(q, n) \in F$ if and only if $q \in F'$, by periodicity of $F$.

Now let $M' = (Q', \Sigma, \delta', 0, F')$. We need to define $\delta'$. But we have an isomorphism $\phi$ between states of $M'$ and states of $M$. We therefore set $\delta' = \phi \circ \delta \circ \phi^{-1}$. That is, $\delta'(q, n, a) = \phi^{-1}(\delta(q + nkT, a))$.

That $M$ and $M'$ accept the same language follows from the isomorphism $\phi$ we have formed between $M$ and $M'$. We know $\delta'(q, n, a) = \phi \circ \delta \circ \phi^{-1}(q, n, a)$ for all $q, n, a$, so $\delta'(0, 0, x) = \phi \circ \delta(0, x)$ for all strings $x$. Since $F' = \phi(F)$, we get that $M$ accepts $x$ if and only if $M'$ accepts $x$, as required.

It remains only to show that $\delta'$ is a valid transition function for $M'$. We must demonstrate that if $\delta'(q, n, a) = (q', n')$ then $|n' - n| \leq 1$. Well,

$$\delta'(q, n, a) = \phi^{-1}(\delta(q + nkT, a))$$

$$= \phi^{-1}(q')$$

where $|q + nkT - q'| \leq C \leq kT$

$$= \phi^{-1}(q'' + n'kT)$$

where $q'' < kT$ and $|n' - n| \leq 1$

$$= (q'', n')$$

where $|n' - n| \leq 1$

as required.

Now choose some CDFA $M = (Q, \Sigma, \delta, q_0, F)$. We wish to construct DPCA $M'$ such that $L(M') = L(M)$.

Let $T = |Q|$. Without loss of generality, take $Q = \{0, 1, \ldots, T - 1\}$ and let $q_0 = 0$. Then we can again define an isomorphism $\phi: Q \times \mathbb{Z} \rightarrow \mathbb{Z}$ by $\phi(q, n) = q + nkT$.

Let $M'$ be the DPCA $(2T, T) \times (\Sigma, \delta', F')$ where $F' = \phi(F \times \mathbb{Z})$. Then $F'$ is indeed $T$-periodic. Define $\delta': \mathbb{Z} \times \Sigma \rightarrow \mathbb{Z}$ by $\delta' = \phi \circ \delta \circ \phi$. As above, that $M$ and $M'$ accept the same language follows directly from the isomorphism between the machines respected by the transition functions and final states.

We must show that $\delta'$ is a valid transition function. That is, we must have $|\delta'(q, a) - q| \leq 2T$ for all $q, a$. Well,

$$\delta'(q, a) = \phi \circ \delta \circ \phi(q, a)$$

$$= \phi(\delta(q \mod T, \lfloor q/T \rfloor, a))$$

$$= \phi(q', n')$$

where $|\lfloor q/T \rfloor - n'| \leq 1$

$$= q' + n'T$$
But now
\[ |\delta'(q, a) - q| = |q' + n'T - q| \]
\[ = |q' + n'T - (q \mod T) + |q/T| T| \]
\[ \leq |q' - (q \mod T)| + ||q/T| T - n'T| \] (Triangle Inequality)
\[ \leq T + T||q/T| - n^i| \]
\[ \leq 2T \]
as required.

4 Example Languages

Theorem 4.1. Let \( L = \{a^i b^i : i \geq 1\} \). Then \( L \in \text{CNTREG} \).

Proof. We construct a CDFA that accepts \( L \). The transition diagram is drawn in Figure 1. We shall briefly describe the operation of this CDFA.

This CDFA uses an implicit error state. State 1 is the start state. Its purpose is to accept \( a^0 b^0 \). From this state we accept some number of \( a \) characters in state 2. For each \( a \) read, the counter is incremented. Note that the value of the counter is irrelevant as we read \( a \) characters. After \( a \) characters are read, we read the \( b \) characters. As long as the counter is not 1, we remain in state 3 and decrement the counter as we read \( b \). When the counter is 1 and we read a \( b \), the counter is decremented to 0 and we move to state 4: the final acceptance state. Since we increment the counter on \( a \) and decrement on \( b \), we must have a string of the form \( a^i b^i \) when we reach state 4. Any further input moves us to the implicit error state, as does any input not of the form \( aa^* b^* \).

Theorem 4.2. Let \( L = \{a^i b^i c^i d^i : i, j \geq 1\} \). Then \( L \not\in \text{CNTREG} \).

Proof. Suppose there exists DPCA\((C, T) M \) that accepts \( L \). Take some non-negative integer \( N \) and consider all strings \( a^i b^j \) where \( i, j < N \). There are \( N^2 \) such strings. Also, given any two unequal strings \( a^i b^j \) and \( a^k b^l \) they are distinguished by suffix \( c^j d^l \) (that is, \( a^i b^j c^j d^l \) is in \( L \), but \( a^k b^l c^j d^l \) is not). Therefore \( \delta(0, a^i b^j) \) must be unique for each \( a^i b^j \). Note that since each transition can span at most \( C \) states, we must have \( |a^i b^j| \leq \delta(0, a^i b^j) \leq C|a^i b^j| \). Therefore, since \( |a^i b^j| \leq 2N \), there are at most \( 2C(2N) = 4CN \) possible values for \( \delta(0, a^i b^j) \). Choose \( N \) large enough that \( N^2 > 4CN \) to arrive at a contradiction.

Theorem 4.3. Let \( L \) be any unary language. Then \( L \in \text{CNTREG} \).
each 2 and 2 imply that \( A \leq \) cannot exist via a density argument.

NOTE: This proof is not yet complete, due to some informal arguments at the end. Any suggestions on a more straightforward method of proof would be greatly appreciated (and credited).

5 Closure Properties

**Theorem 5.1.** CNTREG is closed under negation.

**Proof.** Identical to the proof for DFAs. Take a CDFA and complement the final states.

**Theorem 5.2.** CNTREG is not closed under intersection.

**Proof.** Languages \( \{a^i b^k c^l d^j : i \neq j \} \) are in CNTREG, since we can simply add states to read \( a^* \) and \( d^* \) to a CDFA that accepts \( b^k c^l \) (see Theorem 4.1) and similarly for the second language. However, the intersection of these two languages is \( \{a^i b^k c^l d^j : i \neq j \} \) which we know is not in CNTREG.

**Theorem 5.3.** CNTREG is not closed under concatenation.

**Proof.** NOTE: This proof is not yet complete, due to some informal arguments at the end. Any suggestions on a more straightforward method of proof would be greatly appreciated (and credited).

\( L = \{a^k : k \text{ not a power of 2} \} \) is unary and therefore in CNTREG (Theorem 2). Note that \( L^2 = \{a^i b^j : \text{ neither } i \text{ nor } j \text{ is a power of 2} \} \). It is sufficient to show that \( L^2 \) is not in CNTREG.

Suppose for contradiction there existed a DPCA(\(C,T \)) \(M \) such that \(L(M) = L\). We shall show that \(M\) cannot exist via a density argument.

Choose any two unequal strings \(a^i\) and \(a^j\). There is a unique way to represent \( i \) as \( 2^k - p \) and \( j \) as \( 2^l - q \), where \(0 \leq p < 2^{k-1}\) and \(0 \leq q < 2^{l-1}\). Then neither \(a^i)(a^p)b^3\) nor \(a^j)(a^p)a^2k)b^3\) are in \(L\), since \(2^k - p + p = 2^k\) and \(2^k - q + q + 2^k\) are both powers of 2. However, one of \(a^i)(a^p)b^3\) or \(a^j)(a^p)a^2k)b^3\) must be in \(L\), since if \(2^k - q + q + 2^k\) both powers of 2 then it must be the case that \(2^k - q = 2^k - p\). This would imply \(l = k\) and \(q = p\) by uniqueness, and therefore \(i = j\), a contradiction.

We conclude that \(a^i\) and \(a^j\) are distinguishable by \(M\), and so \(\delta(0,a^i) \neq \delta(0,a^j)\). Let \(A_i = \delta(0,a^i)\) for all \(i\). Then all \(A_i\)s are unique, and since there is a transition from \(A_i\) to \(A_{i+1}\), we must have \(|A_{i+1} - A_i| \leq C\) for each \(i\). Also, since there are infinitely many \(A_i\), \(|A_i|\) must be unbounded. Assume without loss of generality that \(A_i \to +\infty\). Thus any block of \(C\) contiguous positive states must contain at least one \(A_i\).
Now choose any two strings \( a^{i_0}b^{j_0} \) and \( a^{i_1}b^{j_1} \) where neither \( i_0 \) nor \( i_1 \) is a power of 2, and \( j_0 \neq j_1 \). Then these two strings are distinguishable by \( M \). The proof of this is nearly identical to the proof that \( a^i \) and \( a^i \) are distinguishable, so we omit it. Let \( B_{ij} = \delta(a^ib^j) \). Then \( B_{ij} \neq B_{i'j'} \) whenever \( j \neq j' \) and \( i, i' \) are not powers of 2.

Since \( \{A_i\} \) is unbounded and \( |B_{ij} - A_i| \leq jC \), we must have that \( \{B_{ij}\} \) is unbounded as well. For each \( j \), let \( D_j \) be the smallest integer such that any block of \( D_j \) contiguous positive states must contain at least one \( B_{ij} \). We wish to find the value of \( D_j \).

Well, when \( j > 1 \), for any \( i_0 \) and \( i_1 \) there must be a transition from \( B_{i_1(j-1)} \) to \( B_{i_1j} \) and from \( B_{i_0(j-1)} \) to \( B_{i_0j} \). Thus two consecutive \( B_{ij} \) states can be no farther apart than \( 2C \) plus the distance between two \( B_{i_0(j-1)} \) states. Therefore \( D_j \leq D_{j-1} + 2C \).

For \( j = 1 \), recall that every block of \( C \) states must contain at least one \( A_i \), and for sufficiently large \( i \) no two multiples of 2 have a difference less than 3, so any block of \( 3C \) states must contain some \( A_{i_0} \) and \( A_{i_1} \) where \( i_0 \) and \( i_1 \) are not powers of 2. But \( |B_{i_11} - A_{i_0}| \leq C \) and \( |B_{i_11} - A_{i_1}| \leq C \). We conclude that any block of \( 5C \) states must contain at least two states of the form \( B_{i_1} \), so certainly \( D_1 \leq 5C \).

We conclude that \( D_j \leq (3 + 2j)C \) for all \( j \). If we define \( B_{i0} = A_i \) then we get \( D_0 \leq 3C \) as well. But then for any \( N \), a block of \( N \) consecutive positive states must contain at least \( \left\lfloor \frac{N}{D_j} \right\rfloor \) states of the form \( B_{ij} \). These states must all be unique, so we have that

\[
N \leq \sum_j \left\lfloor \frac{N}{(3 + 2j)C} \right\rfloor.
\]

I claim (without proof) that since \( \sum \frac{1}{(3 + 2j)C} \) is divergent, there will exist a sufficiently large \( N \) that (1) will be violated. Heuristically, this is because there is some \( k \) such that \( \sum_{i=0}^{k} \frac{1}{(3 + 2j)C} \rightarrow 1 \). For large enough \( N \) the floor operation will have a negligible impact upon these first \( k \) values. In this case, the sum in (1) will be greater than \( N \). This is a contradiction.

Thus, by contradiction, the machine \( M \) cannot exist.