Note

Two undecidability results for chain code picture languages

Changwook Kim

School of Computer Science, University of Oklahoma, 200 Felgar Street, Norman, OK 73019, USA

Received January 1997; accepted March 2001
Communicated by A. Salomaa

Abstract

It is undecidable whether or not two 1-retreat-bounded regular languages describe exactly the same set of pictures or they describe a picture in common. © 2002 Elsevier Science B.V. All rights reserved.

Keywords: Formal languages; Chain code; Picture languages; Decision properties

1. Introduction

A word over the alphabet \( \Pi = \{u, d, r, l\} \) describes a picture if \( u \) (\( d, r, l \)) denotes a graphics command to draw a unit line in the 2D Cartesian plane by moving the pen upward (downward, rightward, leftward). Such a string is called a chain code and has been used widely for picture encoding and recognition [4, 18]. The chain-code scheme is also similar to the turtle program used in [1], where the trace left by a turtle (or robot) while moving in the plane is interpreted to be a picture. A set of pictures described by chain codes is called a chain code picture language [16]. Mathematical properties of (chain code) picture languages have been studied intensively. Such studies include language-theoretical decision properties [10, 11, 13–15, 17, 20], geometrical/graph-theoretical properties [2, 3], optimizations and transformations [5, 19], and other formal-language theoretical properties [6, 9, 12, 16].

Decision problems for picture languages are hard. For example, there is a regular language for which the picture membership (or recognition) problem is NP-complete [20]. Thus, many restricted classes of picture languages with improved properties (such as better membership complexity or decidable decision problems) have been introduced. Examples include stripe picture languages whose pictures fit into a pair of parallel...

E-mail address: kim@cs.ou.edu (C. Kim).
lines [20], three-way picture languages whose pictures are described by words over a three-letter subset of $\Pi$ [10], and $k$-retreat-bounded picture languages whose pictures are described by making no more than $k$ left moves, ignoring vertical moves, from the rightmost points of any partially drawn picture [11]. See [14, 15] for other restricted types that extend three-way picture languages.

Two major language-theoretic decision problems are the equivalence and intersection-emptiness problems. For picture languages, these problems were shown to be undecidable for regular picture languages in [13] (the equivalence) and in [9, 20] (the intersection emptiness). It is also known that these problems are decidable for 0-retreat-bounded regular picture languages, which are intersection emptiness hold for 2-retreat-bounded regular picture languages. Note that these problems are decidable for 0-retreat-bounded regular picture languages, which are identical to three-way regular picture languages. The purpose of the present paper is to fill the gap for the status of these two decision problems for 1-retreat-bounded picture languages. Thus, it is shown that the equivalence and intersection-emptiness problems are, in fact, undecidable for 1-retreat-bounded regular picture languages.

2. Definitions

For a word $w$ and a symbol $\sigma$, $\#_\sigma(w)$ denotes the number of $\sigma$’s in $w$. The empty word and the empty set are denoted by $\lambda$ and $\emptyset$, respectively. $\Pi$ denotes the picture alphabet $\{u, d, r, l\}$.

Let $\mathcal{Z}$ be the set of all integers. The set $\mathcal{Z} \times \mathcal{Z}$ is called the universal point set and is denoted by $M_0$. For $v = (m, n)$ in $M_0$, the x-component of $v$ is $x(v) = m$ and the y-component of $v$ is $y(v) = n$. The up-, down-, right- and left-neighbors of $v$ are $u(v) = (m, n + 1)$, $d(v) = (m, n - 1)$, $r(v) = (m + 1, n)$ and $l(v) = (m - 1, n)$, respectively. The set $\{(v, \sigma(v)) \mid v \in M_0, \sigma \in \Pi\}$ is called the universal line set and is denoted by $M_1$. For a subset $A$ of $M_1$, its point set is $V(A) = \{v \in M_0 \mid \{v, \sigma\} \in A, \sigma \in \Pi\}$.

An attached picture is a triple $p = (b, s, e)$, where $b$ is a finite subset of $M_1$ and $s, e \in V(b)$ (if $b = \emptyset$ then $s = e \in M_0$); $b$ is called the base (or line set) of $p$, $s$ is called the start point of $p$, and $e$ is called the end point of $p$. The attached pictures obtained by shifting the components of $p$ vertically and/or horizontally form an equivalence class, which is denoted by $\langle p \rangle$ and is referred to as picture.

Each word $w \in \Pi^*$ describes a picture, denoted by $\text{pic}(w)$ and defined inductively by $\text{pic}(\lambda) = \langle \emptyset, (0, 0), (0, 0) \rangle$ and, if $w = z\sigma$ with $z \in \Pi^*$, $\sigma \in \Pi$ and $\text{pic}(z) = \langle b, s, e \rangle$, then $\text{pic}(z\sigma) = \langle b \cup \{e, \sigma(e)\}, s, \sigma(e) \rangle$. Each language $L \subseteq \Pi^*$ describes a picture language, defined by $\text{pic}(L) = \{\text{pic}(w) \mid w \in L\}$. A picture language is regular if it is described by a regular language over $\Pi$.

Let $k$ be a nonnegative integer. A word $w \in \Pi^*$ is a $k$-retreat-bounded word if $\#_u(w') - \#_d(w') \leq k$ for each subword $w'$ of $w$. (This means that $w$ describes its
picture by making no more than \( k \) upward moves, ignoring horizontal moves, from the bottommost points of any partially drawn picture. Thus, we impose the retreat bound vertically, for better readability of the proofs given in the next section, rather than horizontally as in [11].) A language over \( \Pi \) is a \( k \)-retreat-bounded language if it consists of \( k \)-retreat-bounded words; a picture language described by such a language is a \( k \)-retreat-bounded picture language.

3. Main results

We shall prove that the equivalence and intersection-emptiness problems are undecidable for \( 1 \)-retreat-bounded regular picture languages. Many undecidability proofs for picture languages in the literature are based on reductions that simulate Turing machines or linear-bounded automata, the principal technique of which was originally presented in [13]. (See, e.g., [3, 9, 11].) We shall use similar reductions from the emptiness problem for linear-bounded automata, which is undecidable [8]. The two reductions presented in this section are structurally identical; only the picture components for the tape symbols of the given automaton differ, so that a test set appropriate for each case can be easily constructed.

We shall consider the equivalence problem first. Let \( M = (Q, \Sigma, \Gamma, \delta, q_0, @, $, F) \) be an arbitrary linear-bounded automaton with a single tape, where \( Q \) is the set of states, \( \Sigma \) is the input alphabet, \( \Gamma \) is the total tape alphabet, \( \delta : Q \times \Gamma \rightarrow 2^{Q \times \Gamma \times \{l, r\}} \) is the transition function (where \( l \) and \( r \) denote the left and right moves), \( q_0 \in Q \) is the initial state, \( @, $ \in \Gamma \) denote the left and right end-markers of the tape, and \( F \subseteq Q \) is the set of accepting states. We shall assume without loss of generality that \( \Sigma = \{a, b\} \), \( \Gamma = \Sigma \cup \{@, $\} \), \( M \) is an oblivious automaton (which scans @ initially, changes the direction of its head motion only when scans @ or $, and accepts the input word while scanning $), \( M \) never enters \( q_0 \) once it starts computation, and \( M \) traverses its tape at least three times.

Let \( A, B \) and \( C \) be the three pictures shown in Fig. 1(a), where the circle and the square in a picture denote the start point and the end point, respectively, and assume without loss of generality that they also denote any fixed words over \( \Pi \) that describe their corresponding pictures. These picture components are used to simulate a left-to-right traversal of the tape of \( M \), where \( A \) and \( B \) represent the symbols \( a \) and \( b \), respectively, and \( C \) represents both \( @ \) and \( $ \). To simulate a right-to-left traversal of the tape, let \( \tilde{A}, \tilde{B} \) and \( \tilde{C} \) be the pictures obtained from \( A, B \) and \( C \) by switching the start and end points, and assume again that they also denote any fixed words over \( \Pi \) that describe their corresponding pictures.

Let \( h_r \) and \( h_l \) be the homomorphisms (where the subscripts \( r \) and \( l \) denote the right and left moves of the tape head of \( M \)) defined by

\[
h_r : a \mapsto A, \ b \mapsto B,
\]
\[
h_l : a \mapsto \tilde{A}, \ b \mapsto \tilde{B},
\]
and construct a right-linear grammar $G_1 = (Q, \Pi, P_1, q_0)$ such that $P_1$ consists of the following production rules, for all $X, Y \in \{a, b\}$:

- $q_0 \rightarrow Cq$ if $(q, @, r) \in \delta(q_0, @)$,
- $q \rightarrow h_r(X)h_r(Y)q'$ if $(q', Y, r) \in \delta(q, X)$,
- $q \rightarrow CdCq'$ if $(q', @, l) \in \delta(q, @)$,
- $q \rightarrow EUY"$ if $(q', @, r) \in \delta(q, @)$,
- $q \rightarrow C$ if $q \in F$.

It is easy to see that $L(G_1)$ is a 1-retreat-bounded regular language. $G_1$ simulates successive left-to-right and right-to-left traversals of the tape of $M$ by drawing the picture components in Fig. 1(a) in successive rows of the plane. For the simulation of an action of $M$, $G_1$ draws two picture components next to each other, the first being the component for the symbol guessed to be the current tape symbol of $M$ and the second being the component for the symbol rewriting the current tape symbol. This is done by moving rightward (leftward) if the scanned tape symbol is $a$ or $b$ and $M$ moves rightward (leftward) after taking the action, and by moving downward if the scanned tape symbol is $@$ or $\$$. Examples of such pictures drawn by $G_1$ that simulate the tape traversals of $M$ are given in Fig. 1(b), where the first row represents $M$’s action that rewrites the input word $abb$ by $aab$ while traversing the tape from left to right.
right, the second row represents $M$’s action that rewrites $aab$ by $baa$ while traversing the tape from right to left, and so on.

Note that $G_1$ simulates the state transitions of $M$ accurately. Furthermore, for each picture described by $G_1$, the first row represents an accurate simulation of the initial left-to-right traversal of the tape of $M$. Then, it is not difficult to see that, for each word $w \in \{a, b\}^*$, $w \in L(M)$ if and only if there exists a word $\tilde{w} \in L(G_1)$ such that (i) the picture components for $a$ and $b$ in the even-numbered columns of the first row of $\text{pic}(\tilde{w})$ represent the symbols of $w$, and (ii) the rest of $\text{pic}(\tilde{w})$ is drawn in such a way that every picture component in the $i$th row, $i \geq 2$, that corresponds to reading a tape symbol (which is guessed by $G_1$) matches correctly with the actual tape symbol, the picture component of which is located immediately above the guessed component, in the $(i-1)$th row. Observe that, in Fig. 1(b), $G_1$ makes correct guessings of the tape symbols in the second traversal of the tape but makes a wrong guessing in each of the third through fifth traversals (see the second, seventh, and sixth columns).

The condition stated in (ii) above is equivalent to the following: for each $i \geq 2$, if $i$ is an even (odd) number then the picture component located in each odd (even) numbered column of the $i$th row is identical to the picture component located immediately above it in $\text{pic}(\tilde{w})$. We shall construct a test set $L(T_1)$, defined by a regular expression $T_1$, which describes all pictures violating this condition. Let $\Omega = \{ud, d, r, l\}$. For each word $w \in \Pi^*$, with $\text{pic}(w) = \langle \langle b, (0,0), e \rangle, \xi x \rangle = x(e)$ and $\xi y (w) = y(e)$. Let

$$E = \{ w \in \Omega^* | \xi x (w) = 0 \pmod{8}, \xi y (w) = 0 \pmod{2} \} ,$$

$$O = \{ w \in \Omega^* | \xi x (w) = 0 \pmod{8}, \xi y (w) = 1 \pmod{2} \} .$$

It is easy to see that $E$ and $O$ are 1-retreat-bounded regular languages. Let us assume without loss of generality that any fixed regular expressions for these two sets are also denoted by $E$ and $O$, respectively. Let $F$ denote a regular expression for $\Omega^*$. Now, define

$$T_1 = E(Ad(B + C) + Bd(A + C) + Cd(A + B))F + O(Ad(B + C) + Bd(A + C) + Cd(A + B))F .$$

$L(T_1)$ is clearly a 1-retreat-bounded regular language. Then, $L(G_1) \cup L(T_1)$ is also a 1-retreat-bounded regular language. It is straightforward to observe that $L(M) = \emptyset$ if and only if $\text{pic}(L(G_1)) \subseteq \text{pic}(L(T_1))$ if and only if $\text{pic}(L(G_1) \cup L(T_1)) = \text{pic}(L(T_1))$. As the problem “$L(M) = \emptyset$” is undecidable for a linear-bounded automaton $M$, it follows that the following theorem holds.

**Theorem 3.1.** It is undecidable whether or not $\text{pic}(L_1) = \text{pic}(L_2)$ for 1-retreat-bounded regular languages $L_1$ and $L_2$.

We shall consider now the intersection-emptiness problem. For this, let $M$ be a linear-bounded automaton with its definition as given before. Let $A_r$, $B_r$, $C_r$, $A_w$, $B_w$ and $C_w$ be the six pictures shown in Fig. 2(a). These picture components will be used
to simulate a left-to-right traversal of the tape of $M$, where $A$, $B$ and $C$ represent the tape symbols $a$, $b$ and @,$\$ as before and the subscripts $r$ and $w$ indicate reading and writing of the symbols. (Thus, we use different picture components for reading and writing here.) Now, let $\overline{A}_r$, $\overline{B}_r$, $\overline{C}_r$, $\overline{A}_w$, $\overline{B}_w$ and $\overline{C}_w$ be the pictures obtained from these six pictures by switching the start and end points, which will be used to simulate a right-to-left traversal of the tape of $M$. As before, we shall assume that these symbols denote also any fixed words over $\Pi$ that describe their corresponding pictures.

Let $h^r_r$, $h^w_r$, $h^r_l$ and $h^w_l$ be the homomorphisms (where the subscripts $r$ and $l$ denote the right and left moves as in $h_r$ and $h_l$ defined for the equivalence case and the superscripts $r$ and $w$ denote reading and writing of the tape symbols) such that

- $h^r_r: a \mapsto \overline{A}_r$, $b \mapsto \overline{B}_r$,
- $h^w_r: a \mapsto \overline{A}_w$, $b \mapsto \overline{B}_w$,
- $h^r_l: a \mapsto \overline{A}_r$, $b \mapsto \overline{B}_r$,
- $h^w_l: a \mapsto \overline{A}_w$, $b \mapsto \overline{B}_w$. 

Fig. 2. Picture components used by $G_2$ and examples of pictures drawn by $G_2$ to simulate the tape traversals of $M$. 

Construct a right-linear grammar \( G_2 = (Q, \Pi, P_2, q_0) \) such that \( P_2 \) consists of the following production rules, for all \( X, Y \in \{a, b\} \):

\[
\begin{align*}
q_0 & \rightarrow Cwq & \text{if } (q, @, r) \in \delta(q_0, @), \\
q & \rightarrow h_r'(X)h_r''(Y)q' & \text{if } (q', Y, r) \in \delta(q, X), \\
q & \rightarrow C_dCwq' & \text{if } (q', @, r) \in \delta(q, @), \\
q & \rightarrow hr_l(X)hw_l(Y)q' & \text{if } (q', Y, l) \in \delta(q, X), \\
q & \rightarrow C_r & \text{if } q \in F.
\end{align*}
\]

\( L(G_2) \) is clearly a 1-retreat-bounded regular language. \( G_2 \) simulates \( M \) similarly to \( G_1 \); it only uses different picture components. Examples of pictures drawn by \( G_2 \) that simulate the tape traversals of \( M \) are given in Fig. 2(b), where the first row represents \( M \)'s action that rewrites \( aaba \) by \( abab \), the second row represents \( M \)'s action that rewrites \( abab \) by \( baba \), and so on. As in the equivalence case, \( G_2 \) simulates \( M \)'s action accurately if the tape symbol guessed by \( G_2 \) is always equivalent to the actual tape symbol, whose corresponding picture component is located immediately above the picture component for the former. It is not difficult to observe that this condition is satisfied if and only if the picture drawn by \( G_2 \) is completely filled horizontally except for the topmost and bottommost horizontal line segments. For example, in Fig. 2(b), \( G_2 \) makes correct guessings of the tape symbols in the second row but makes a wrong guessing in each of the third through fifth rows.

Let \( I = urdlr \), \( J = udrud \), \( \bar{I} = uldrl \), \( \bar{J} = drulr \) and \( \bar{J} = durdu \). Then, the test set for \( G_2 \) is defined by the following regular expression:

\[
T_2 = (I + J)^+ (dI^+ dI^+)^+ dI^+ (\bar{I} + \bar{J})^+ d.
\]

It is straightforward to observe that \( L(T_2) \) is a 1-retreat-bounded regular language and, furthermore, \( L(M) = \emptyset \) if and only if \( L(G_2) \cap L(T_2) = \emptyset \). Therefore, the following theorem holds.

**Theorem 3.2.** It is undecidable whether or not \( \text{pic}(L_1) \cap \text{pic}(L_2) = \emptyset \) for 1-retreat-bounded regular languages \( L_1 \) and \( L_2 \).

In the literature, the pictures considered in this paper are called drawn pictures and their line sets only (with no start and end points specified) are called basic pictures. It is a trivial matter to check that the proofs given in this section work equally well for basic 1-retreat-bounded regular picture languages.

**References**
