A REMARK ON TWO-DIMENSIONAL 
FINITE AUTOMATA

Akira Nakamura
Department of Applied Mathematics
Hiroshima University
and
Computer Science Center
University of Maryland
College Park, MD 20742

ABSTRACT

Let S2-APMOTA(m) be an area-preserving two-dimensional multipass on-line tessellation acceptor over square array input languages whose pass number is bounded by m. It is proved that an open problem "Is \( L(2-NA) \subseteq L(S2-AMPOTA(1)) \)?" proposed in a previous paper by Inoue and the present author has a positive solution.

The support of the U.S. Air Force Office of Scientific Research under Grant AFOSR-77-3271 is gratefully acknowledged, as is the help of Kathryn Riley in preparing this paper.
In [1], we showed that the class of sets accepted by nondeterministic two-dimensional on-line tessellation acceptors properly contains that accepted by two-dimensional nondeterministic finite automata (i.e., \( L(2-NA) \subseteq L(2-OTA) \)) and also that \( L(2-DOTA) \) is incomparable with \( L(2-NA) \) and \( L(2-DA) \). This result (Theorem 4.1 in [1]) was the main theorem of [1]. Also we defined in [2] an area-preserving two-dimensional multipass on-line tessellation acceptor (i.e., 2-AMPOTA) and defined in [3] a 2-AMPOTA whose pass number is bounded by \( m \) (i.e., 2-AMPOTA(\( m \))). Further, the 2-AMPOTA(\( 1 \)) over square array input languages was denoted by \( S2-APMOTA(1) \). In this notation, we proposed in [3] an open problem "Is \( L(2-NA) \subseteq L(S2-AMPOTA(1)) \)?".

In [1], the inputs were rectangular array languages. In this note, we consider square array languages. We prove that Theorem 4.1 in [1] is also valid for square array languages, and hence show that the open problem in [3] has a positive solution.

Let us consider a set \( \Sigma = \{0,1,c,e,\#\} \) of input symbols and also the input square array languages surrounded by the special boundary symbol \( \# \). In this note, we treat exclusively input languages such as shown in Fig. 1.
Figure 1

In Fig. 1, \( a_{ij} \) is 0 or 1.

Now, we consider as a **chunk** a part of the form

as shown in Fig. 2:
Figure 2

Let us denote the diagonal parts of the chunk by $p_1$ and $p_2$ respectively (Fig. 3):
This chunk plays the same role as the chunk in [1], and the diagonals $p_1$ and $p_2$ correspond to the rows $r_1$ and $r_2$, respectively.

A chunk as in Fig. 4 is called an $(\ell,n)$-chunk.
Let $M$ be a 2-NA and $x, y$ be any different $(\ell, n)$-chunks. Then $M$-equivalence of $x$ and $y$ is defined in a similar way as in [1]. Note that $M$ always enters or exits a chunk at the diagonals $p_1$ or $p_2$. 
Now, let us consider a set $T$ of pictures such as shown in Fig. 1 in which there exists some $i$ ($1 \leq i \leq \ell$) such that $a_{i_1}a_{i_2}\ldots a_{i_n}$ is the same as the head $a_0a_1a_2\ldots a_{n}$ of the top row.

**Theorem 1.**

1. $T \in L(2$-DOTA$)$
2. $T \notin L(2$-NA$)$.

**Proof:**

(1) is shown without difficulty.

(2) is provable as follows:

By the same considerations as in [1], there are at most

$$s = (2^{2(n+2)}k^n + 1)2^{2(n+2)}k$$

$\mathcal{M}$-equivalence classes of $(2^n,n)$-chunks, where $k$ is the number of states of $M$. We denote those classes by $C_1, C_2, \ldots, C_s$.

There are $2^n$ different strings over $(0,1)$ of length $n$. Let us denote those strings by $R_1, R_2, \ldots, R_{2^n}$. Here, we distinguish chunks depending on the appearances of these rows. For example, $[R_1]$ means a chunk in which only rows corresponding to $R_1$ appear (see Fig. 5).
[$R_1, R_2$] means a chunk in which only rows corresponding to $R_1$ and $R_2$ appear (see Fig. 6).

![Figure 6](image)

According to this characterization we know that there are $\nu$ different languages for $(2^n, n)$-chunks, where

$$\nu = \frac{2^n C_1^+}{2^n C_2^+} \cdots + \frac{2^n C_n}{2^n} = 2^{2n}.$$ 

Since $\nu > 0$ for large $n$, we know the following fact: There exists some $R_j$ such that $R_j$ appears in some chunk $C_0$ but does not appear in some chunk $C_0'$ where $C_0'$ is $M$-equivalent to $C_0$. Thus, we can prove that $T \in L(2-NA)$ implies a contradiction by making use of the definition of $M$-equivalence of chunks. Therefore, we get (2)./

Theorem 2. $L(2-NA) \subseteq L(S2-AMPOTA(1))$.

Proof:

$L(S2-AMPOTA(1))$ is the same as $L(2-OTA)$ over the square array.
Thus, it is provable in the same way as in [1] that an
$S_2$-$\text{AMPOTA}(1)$ can simulate a $2$-$\text{NA}$. Therefore, from Theorem
1 this theorem follows.//

From Theorem 1, we can prove the incomparability of
$L(S_2$-$\text{DAMPOTA}(1))$ with $L(2$-$\text{NA})$ and $L(2$-$\text{DA})$ by the same method
as in [1].
References


Acknowledgement

The author would like to thank Professor Azriel Rosenfeld for arranging a very enjoyable four week stay at the Computer Science Center in July 1979, and for valuable comments.
Let $S2$-APOTA$(m)$ be an area-preserving two-dimensional multipass on-line tessellation acceptor over square array input languages whose pass number is bounded by $m$. It is proved than an open problem: Is $L(2-NA) \subseteq L(S2$-APOTA$(1))$? proposed in a previous paper by Inoue and the present author has a positive solution.