Two-Dimensional Alternating Turing Machines with Only Universal States

AKIRA ITO, KATSUSHI INOUE, ITSUO TAKANAMI, AND HIROSHI TANIGUCHI

Department of Electronics, Faculty of Engineering, Yamaguchi University, Ube 755, Japan

Several properties of two-dimensional alternating Turing machines are investigated. The first part of this paper investigates the relationship between the classes of sets accepted by space-bounded and finitely leaf-size bounded three-way two-dimensional alternating Turing machines and the classes of sets which are finite intersections of sets accepted by space-bounded three-way two-dimensional nondeterministic Turing machines. The second part of this paper investigates the accepting power and closure properties (under Boolean operations) of two-dimensional alternating Turing machines with only universal states.

1. INTRODUCTION

Alternating Turing machines were introduced in Chandra et al. (1981) as a generalization of nondeterministic Turing machines and as a mechanism to model parallel computation. In related papers (Ladner et al., 1978; Ruzzo, 1980; Paul et al., 1980; Paul and Reischuk, 1980; King, 1980, 1981; Sudborough, 1980; Gurari and Ibarra, 1982), investigations of alternating machines have continued. Many problems about alternating machines remain to be solved, however.

In Inoue et al. (1982, 1983), the authors introduced two-dimensional alternating Turing machines, and gave several properties of these machines. This paper continues the investigation of fundamental properties of two-dimensional alternating Turing machines, whose input tapes are restricted to square ones. We also investigate some properties of three-way two-dimensional alternating Turing machines. The three-way restriction is the two-dimensional analogue of the one-way restriction for one-dimensional machines studied by Hopcroft and Ullman (1967) and others. We believe that these investigations will give the reader deep understanding of two-dimensional Turing machines and alternating Turing machines.

Section 2 gives terminology and notation necessary for this paper.

For each integer $k \geq 1$, and any space-bounded function $L$, let
\( L[\text{TAT}(L(m), k)] \) (resp. \( L[\text{TNT}(L(m))] \)) be the class of sets of square tapes accepted by \( L(m) \) space-bounded and \( k \) leaf-size bounded three-way two-dimensional alternating Turing machines (Inoue et al., 1982, 1983) (resp. accepted by \( L(m) \) space-bounded three-way two-dimensional nondeterministic Turing machines (Inoue and Takanami, 1979)). Further, for each integer \( k \geq 1 \), let \( L_k[\text{TNT}(L(m))] = \{ A_1 \cap \cdots \cap A_k \mid \text{each } A_i \in L[\text{TNT}(L(m))] \} \).

It seems interesting to us to investigate the relationship between \( L[\text{TAT}(L(m), k)] \) and \( L^k[\text{TNT}(L(m))] \). The main purpose of Section 3 is to show that \( L^k[\text{TNT}(L(m))] \not\subseteq L[\text{TAT}(L(m), k)] \) for any integer \( k \geq 2 \) and any \( L \) such that \( L(m) \ll \log m \) where for functions \( f(m) \) and \( g(m) \), we write \( f(m) \sim g(m) \) when \( \lim_{m \to \infty} \frac{f(m)}{g(m)} = 0 \).

From this result, we might say that a parallel machine with cooperative processors is in general more powerful than a mechanism with the same number of processors which run independently.

Sections 4, 5, and 6 investigate fundamental properties of two-dimensional alternating Turing machines which have only universal states, and whose input tapes are restricted to square ones. It is a very important unsolved problem (Chandra et al., 1981) whether \( \text{ALTIP} \) (the class of sets accepted by one-dimensional alternating Turing machines with only universal states in polynomial time) is equal to \( \text{NPTIME} \) (the class of sets accepted by one-dimensional nondeterministic Turing machines in polynomial time). We cannot solve this problem, but in Section 4 we show that there exists a set accepted by \( L(m) \) space-bounded three-way two-dimensional nondeterministic Turing machines, but not accepted by any \( L(m) \) space-bounded three-way two-dimensional alternating Turing machine with only universal states (\( \text{TUT}(L(m)) \)), for any \( L \) such that (i) \( L(m) \ll m \), or (ii) \( L(m) \gg \log m \) and \( L(m) \ll m^2 \). From this result, for example, we may say that for any \( L \) such that \( L(m) \ll m^2 \), \( \text{TUT}(L(m)) \)s are less powerful than ordinary \( L(m) \) space-bounded three-way two-dimensional alternating Turing machines (\( \text{TAT}(L(m)) \)s). In Section 5, we investigate the accepting power of four-way two-dimensional alternating Turing machines, and show, for example, that for any \( L \) such that \( L(m) \ll \log m \), \( L(m) \) space-bounded four-way two-dimensional alternating Turing machines with only universal states (\( \text{UT}(L(m)) \)s) are less powerful than ordinary \( L(m) \) space-bounded four-way two-dimensional alternating Turing machines (\( \text{AT}(L(m)) \)s). Section 6 investigates closure properties of \( \text{TUT}(L(m)) \)s, where \( L(m) \ll m^2 \), under Boolean operations. We show, for example, that for any \( L \) such that (i) \( L(m) \ll m \), or (ii) \( L(m) \gg \log m \) and \( L(m) \ll m^2 \), the class of sets accepted by \( \text{TUT}(L(m)) \)s is not closed under complementation.

We conclude this section by summarizing the main results of this paper. For any function \( L \), let \( L[\text{TDT}(L(m))] \) (resp. \( L[\text{DT}(L(m))] \)) be the class of sets of square tapes accepted by \( L(m) \) space-bounded three-way two-dimensional alternating Turing machines (Inoue et al., 1982, 1983) (resp. \( L(m) \) space-bounded three-way two-dimensional nondeterministic Turing machines (Inoue and Takanami, 1979)). Further, for each integer \( k \geq 1 \), let \( L_k[\text{TNT}(L(m))] = \{ A_1 \cap \cdots \cap A_k \mid \text{each } A_i \in L[\text{TNT}(L(m))] \} \).
dimensional deterministic Turing machines (resp. accepted by $L(m)$ space-bounded four-way two-dimensional deterministic Turing machines).

1. If $L(m) \leq \log m$, then $L^{k} [TNT(L(m))] \subseteq L [TAT(L(m), k)]$ for any $k \geq 2$ (Theorem 3.3).

2. If (i) $L(m) \leq m$, or (ii) $L(m) \geq \log m$ and $L(m) \leq m^2$, then $L [TDT(L(m))] \subseteq L [TUT(L(m))] \subseteq L [TAT(L(m))]$, and $L [TNT(L(m))]$ and $L [TUT(L(m))]$ are incomparable (Theorem 4.1).

3. If (i) $L(m) \leq m$, or (ii) $L(m) \geq \log m$ and $L(m) \leq m^2$, then $L [TUT(L(m))] \subseteq L [UT(L(m))] \subseteq L [TAT(L(m))]$, and $L [UT(L(m))]$ and $L [TNT(L(m))]$ are incomparable (Theorem 5.2).

4. If $L(m) \leq \log m$, then $L [DT(L(m))] \subseteq L [UT(L(m))] \subseteq L [TAT(L(m))]$ (Theorem 5.1).

5. $L [TUT(0)] \subseteq L [TDT(m)]$, and space $m$ is necessary for TDTs to simulate TUT(0)s (Theorem 4.2).

6. $L [UT(0)] \subseteq L [TDT(m^2)]$, and space $m^2$ is necessary for TDTs to simulate UT(0)s (Theorem 5.3).

7. $L [TUT(L(m))]$ is closed under intersection (Theorem 6.6), but not under union if $L(m) \leq \log m$ (Theorem 6.5), nor under complementation if (i) $L(m) \leq m$, or (ii) $L(m) \geq \log m$ and $L(m) \leq m^2$ (Theorem 6.1).

2. Preliminaries

Definition 2.1. Let $\Sigma$ be a finite set of symbols. A two-dimensional tape over $\Sigma$ is a two-dimensional rectangular array of elements of $\Sigma$.

The set of all two-dimensional tapes over $\Sigma$ is denoted by $\Sigma^{(2)}$. Given a tape $x \in \Sigma^{(2)}$, we let $l_1(x)$ be the number of rows of $x$ and $l_2(x)$ be the number of columns of $x$. The set of all two-dimensional tape $x$ over $\Sigma$ such that $l_1(x) = m$ and $l_2(x) = n$ is denoted by $\Sigma^{m \times n}$. If $1 \leq i \leq l_1(x)$ and $1 \leq j \leq l_2(x)$, we let $x(i, j)$ denote the symbol in $x$ with coordinates $(i, j)$. Furthermore, we define

$x[(i, j), (i', j')]$,

only when $1 \leq i \leq i' \leq l_1(x)$ and $1 \leq j \leq j' \leq l_2(x)$, as the two-dimensional tape $z$ satisfying the following:

(i) $l_1(z) = i' - i + 1$ and $l_2(z) = j' - j + 1$;

(ii) for each $k, r$ [$1 \leq k \leq l_1(z)$, $1 \leq r \leq l_2(z)$], $z(k, r) = x(k + i - 1, r + j - 1)$. (We call $x[(i, j), (i', j')]$ the $(i, j), (i', j')]$-segment of $x$.) For $x \in \Sigma^{m \times n}$ the $i$th row $x[(i, 1), (i, n)]$ of $x$ is simply denoted by $x[i, \ast]$.

This paper assumes that the reader is familiar with two-dimensional
Turing machines (Inoue and Takanami, 1979) and one-dimensional alternating automata (Chandra et al., 1981, Ladner et al., 1978).

Two-dimensional alternating Turing machines were introduced in Inoue et al. (1982, 1983).

**Definition 2.2.** A two-dimensional alternating Turing machine (AT) is a seven-tuple

\[ M = (Q, q_0, U, F, \Sigma, \Gamma, \delta), \]

where

1. \( Q \) is a finite set of states,
2. \( q_0 \in Q \) is the initial state,
3. \( U \subseteq Q \) is the set of universal states,
4. \( F \subseteq Q \) is the set of accepting states,
5. \( \Sigma \) is a finite input alphabet (# \( \in \Sigma \) is the boundary symbol),
6. \( \Gamma \) is a finite storage tape alphabet (\( B \in \Gamma \) is the blank symbol),
7. \( \delta \subseteq (Q \times (\Sigma \cup \{\#\}) \times \Gamma) \times (Q \times (\Gamma - \{B\}) \times \{\text{left, right, up, down, no move}\}) \) is the next move relation.

A state \( q \) in \( Q - U \) is said to be existential. As shown in Fig. 1, the machine \( M \) has a read-only (rectangular) input tape with boundary symbols "#" and one semi-infinite storage tape\(^1\) initially blank. Of course, \( M \) has a finite control, an input tape head, and a storage tape head. A position is assigned to each cell of the read-only input tape and to each cell of the storage tape, as shown in Fig. 1. A step of \( M \) consists of reading one symbol from each tape, writing a symbol on the storage tape, moving the input and storage heads in specified directions, and entering a new state, in accordance with the next move relation \( \delta \). Note that the machine cannot write the blank symbol. If the input head falls off the input tape, or if the storage head falls off the storage tape (by moving left), then the machine \( M \) can make no further move.

**Definition 2.3.** A configuration of an AT \( M = (Q, q_0, U, F, \Sigma, \Gamma, \delta) \) is a pair of an element of \( \Sigma^{(2)} \) and an element of

\[ C_M = (N \cup \{0\})^2 \times S_M, \]
\[ S_M = Q \times (\Gamma - \{B\})^* \times N, \]

where \( N \) denotes the set of all positive integers. The first component of a

\(^1\) The restriction to one semi-infinite storage tape entails no loss of generality, since we consider only space-bounded machines.
configuration\(^2\) \(c = (x, ((i,j), (q, a, k)))\) represents the input to \(M\). The first component \((i,j)\) of the second component of \(c\) represents the input head position, and the second component \((q, a, k)\) of the second component of \(c\) represents the state of the finite control, nonblank contents of the storage tape, and the storage head position. An element of \(C_M\) is called a semi-configuration of \(M\) and an element of \(S_M\) is called a storage state of \(M\). If \(q\) is the state associated with configuration \(c\), then \(c\) is said to be a universal (existential, accepting) configuration if \(q\) is a universal (existential, accepting) state. The initial configuration of \(M\) on input \(x\) is

\[
I_M(x) = (x, ((1, 1), (q_0, \lambda, 1))),
\]

where \(\lambda\) is the null string.

A configuration represents an instantaneous description of \(M\) at some point in a computation.

**Definition 2.4.** Given \(M = (Q, q_0, U, F, \Sigma, \Gamma, \delta)\), we write \(c \vdash_M c'\) and say \(c'\) is a successor of \(c\) if configuration \(c'\) follows from configuration \(c\) in one step of \(M\), according to the transition rules \(\delta\). The relation \(\vdash_M\) is not necessarily single valued, since \(\delta\) is not. \(\vdash_M^\ast\) denotes the reflexive transitive closure of \(\vdash_M\). A computation path of \(M\) on \(x\) is a sequence \(c_0 \vdash_M c_1 \vdash_M \cdots \vdash_M c_n\) \((n \geq 0)\), where \(c_0 = I_M(x)\). A computation tree of \(M\) is a finite, nonempty labeled tree with the properties

\(^2\) We denote that \(0 \leq i \leq l_1(x) + 1\), \(0 \leq j \leq l_2(x) + 1\), and \(1 \leq k \leq |a| + 1\), where for any string \(w\), \(|w|\) denotes the length of \(w\) (with \(|\lambda| = 0\).
(1) each node $\pi$ of the tree is labeled with a configuration $l(\pi)$,

(2) if $\pi$ is an internal node (a non-leaf) of the tree, $l(\pi)$ is universal and \( \{c | l(\pi) \vdash_M c\} = \{c_1, \ldots, c_k\} \), then $\pi$ has exactly $k$ children $\rho_1, \ldots, \rho_k$ such that $l(\rho_i) = c_i$.

(3) if $\pi$ is an internal node of the tree and $l(\pi)$ is existential, then $\pi$ has exactly one child $\rho$ such that $l(\pi) \vdash_M l(\rho)$.

An accepting computation tree of $M$ on an input $x$ is a computation tree whose root is labeled with $I_M(x)$ and whose leaves are all labeled with accepting configurations. We say that $M$ accepts $x$ if there is an accepting computation tree of $M$ on input $x$. Define

$$T(M) = \{x \in \Sigma^2 | M \text{ accepts } x\}.$$ 

We next recall the definition of a three-way two-dimensional alternating Turing machine (Inoue et al., 1982, 1983).

**Definition 2.5.** A three-way two-dimensional alternating Turing machine (TAT) is an AT $M = (Q, q_0, U, F, \Sigma, \Gamma, \delta)$ such that

$$\delta \subseteq (Q \times (\Sigma \cup \{\#\}) \times \Gamma) \times (Q \times (\Gamma - \{B\}) \times \{\text{left, right, down, no move}\} \times \{\text{left, right, no move}\}).$$

(That is, a TAT is an AT whose input head can move left, right, or down, but not up.)

In this paper, we shall investigate the properties of ATs and TATs whose input tapes are restricted to square ones and whose storage tapes are bounded (in length) to use.

Let $L : N \rightarrow R$ be a function with one variable $m$, where $R$ denotes the set of all nonnegative real numbers. With each AT (or TAT) $M$ we associate a space complexity function $SPACE$ which takes configurations to natural numbers. That is, for each configuration $c = (x, ((i,j), (q, a, k)))$, let $SPACE(c) = \alpha^0$. We say that $M$ is $L(m)$ space-bounded if for all $m$ and for all $x$ with $l_1(x) = l_2(x) = m$, if $x$ is accepted by $M$ then there is an accepting computation tree of $M$ on input $x$ such that for each node $\pi$ of the tree, $SPACE(l(\pi)) \leq [L(m)]^3$. By "AT($L(m)$)" ("TAT($L(m)$)") we denote an $L(m)$ space-bounded AT (TAT) whose input tapes are restricted to square ones.\(^4\) Define

$$\mathcal{L}[\text{AT}(L(m))] = \{T | T = T(M) \text{ for some AT}(L(m)) M\},$$

$$\mathcal{L}[\text{TAT}(L(m))] = \{T | T = T(M) \text{ for some TAT}(L(m)) M\}.$$ 

\(^3\) $[r]$ means the smallest integer greater than or equal to $r$.

\(^4\) In Inoue et al. (1982, 1983), an $L(m)$ space-bounded AT (TAT) whose input tapes are restricted to square ones is denoted by 2-ATM\(^3\)(L(m)) (TR2-ATM\(^3\)(L(m))).
By using the well-known technique, it is easily proved that for any constant \( k \geq 0 \), \( \mathcal{L}[AT(k)] = \mathcal{L}[AT(0)] \) and \( \mathcal{L}[TAT(k)] = \mathcal{L}[TAT(0)] \). In particular, we denote an \( AT(0) \) (\( TAT(0) \)) by "AF" ("TAF"). An AF (TAF) is an alternating version of a two-dimensional finite automaton (Inoue and Nakamura, 1977; Rosenfeld, 1979; Blum and Hewitt, 1967) (three-way two-dimensional finite automaton (Inoue and Takanami, 1979; Rosenfeld, 1979) whose input tapes are restricted to square ones.

Two-dimensional deterministic and nondeterministic Turing machines (three-way two-dimensional Turing machines) (Inoue and Takanami, 1979) are special cases of ATs (TATs). For example, a two-dimensional nondeterministic Turing machine is an AT which has no universal state, and a two-dimensional deterministic Turing machine is an AT whose configurations each have at most one successor. By \( NT(L(m)) \) (\( DT(L(m)) \), \( TNT(L(m)) \), \( TDT(L(m)) \)) we denote an \( L(m) \) space-bounded\(^5\) two-dimensional nondeterministic Turing machine (two-dimensional deterministic Turing machine, three-way two-dimensional nondeterministic Turing machine, three-way two-dimensional deterministic Turing machine) with square input tapes. (See Inoue and Takanami, 1979 for definitions of these machines.) Furthermore, by \( NF \) (\( DF \), \( TNF \), \( TDF \)) we denote a two-dimensional nondeterministic finite automaton (two-dimensional deterministic finite automaton, three-way two-dimensional nondeterministic finite automaton, three-way two-dimensional deterministic finite automaton) with square input tapes. (See Inoue and Takanami, 1979; Blum and Hewitt, 1967 for definitions of these automata.) Let \( \mathcal{L}[NT(L(m))] = \{T | T = T(M) \text{ for some } NT(L(m)) M \} \), \( \mathcal{L}[DT(L(m))] \), \( \mathcal{L}[NF] \), etc., are defined similarly.

### 3. Accepting Power of Leaf-Size Bounded TATs

In Inoue et al. (1982, 1983), we introduced a new concept, called "leaf-size bounded computation," of alternating Turing machines.\(^6\)

Basicallly, the "leaf-size" used by a TAT on a given input is the number of leaves of an accepting computation tree with fewest leaves. Leaf-size, in a sense, reflects the minimum number of processors that run in parallel in accepting a given input.

**Definition 3.1.** Let \( Z : N \rightarrow R \) be a function. For each tree \( t \), let \( LEAF(t) \) denote the leaf-size of \( t \) (i.e., the number of leaves of \( t \)). We say that a TAT \( M \) is \( Z(m) \) leaf-size bounded if for all \( x \) with \( l_1(x) = l_2(x) = m \), if

\(^5\) In Inoue and Takanami (1979), the term "\( L(m) \) tape-bounded" is used instead of the term "\( L(m) \) space-bounded."

\(^6\) King (1981) independently introduced the same complexity measure as "leaf-size." In King (1981), the term "branching" is adopted instead of the term "leaf-size."
x is accepted by M then there is an accepting computation tree t of M on x such that LEAF(t) \leq |Z(m)|.

By TAT(L(m), Z(m)), we denote a Z(m) leaf-size bounded TAT(L(m)). Let TAF(Z(m)) denote a Z(m) leaf-size bounded TAF. Define

\[ \mathcal{L}[\text{TAT}(L(m), Z(m))] = \{ T | T = T(M) \text{ for some } \text{TAT}(L(m), Z(m)) M \}. \]

\[ \mathcal{L}[\text{TAF}(Z(m))] \] is defined similarly.

**Definition 3.2.** For any integer \( k \geq 1 \) and for any \( L : N \rightarrow R \), define

\[ \mathcal{L}^k[\text{TNT}(L(m))] = \begin{cases} \mathcal{L}[\text{TNT}(L(m))] & \text{if } k = 1, \\ \{ A_1 \cap \cdots \cap A_k | A_i \in \mathcal{L}[\text{TNT}(L(m))], 1 \leq i \leq k \} & \text{otherwise.} \end{cases} \]

\[ \mathcal{L}^k[\text{TNT}], \text{ etc., are defined similarly.} \]

It will be interesting to investigate a relationship between \( \mathcal{L}[\text{TAT}(L(m), k)] \) and \( \mathcal{L}^k[\text{TNT}(L(m))] \). The main purpose of this section is to show that \( \mathcal{L}^k[\text{TNT}(L(m))] \subseteq \mathcal{L}[\text{TAT}(L(m), k)] \) for any integer \( k \geq 2 \) and any \( L : N \rightarrow R \) such that \( L(m) \leq \log m \).

We first show that \( \mathcal{L}^k[\text{TNT}(L(m))] \subseteq \mathcal{L}^{k+1}[\text{TNT}(L(m))] \) for any \( k \geq 1 \) and any \( L \) such that \( L(m) \leq \log m \).

**Lemma 3.1.** For any integer \( k \geq 1 \) and for any \( L : N \rightarrow R \),

\[ \mathcal{L}^k[\text{TNT}(L(m))] \subseteq \mathcal{L}[\text{TAT}(L(m), k)]. \]

**Proof.** For \( k = 1 \), the lemma is obvious. For any \( k \geq 2 \), let \( A \) in \( \mathcal{L}^k[\text{TNT}(L(m))] \) be equal to \( A_1 \cap \cdots \cap A_k \), where for each \( i \) (\( 1 \leq i \leq k \)), \( A_i \) is accepted by some \( \text{TNT}(L(m)) M_i \). We consider a \( \text{TAT}(L(m), k) M' \) which acts as follows. Given an input \( x \), \( M' \) first universally branches to directly simulate the actions of \( M_1, \ldots, M_{k-1} \), and \( M_k \) on \( x \). \( M' \) enters an accepting state if an only if \( M_i, 1 \leq i \leq k, \) enters an accepting state. It is obvious that \( T(M') = A \). Q.E.D.

**Lemma 3.2.** For each integer \( k \geq 1 \), let \( T[k] = \{ x \in \{0, 1\}^{m \times m} | m \geq k \& (\text{there exist exactly } k \text{ 1's on the first row of } x) \& x[1, *] = x[2, *] \} \). Then

1. \( T[k] \in \mathcal{L}^k[\text{TDF}] \);
2. \( T[k+1] \notin \mathcal{L}^k[\text{TNT}(L(m))] \) for any \( L : N \rightarrow R \) such that \( L(m) \leq \log m \).
Proof. (1) For each $k \geq 1$ and each $r(1 \leq r \leq k)$, let $T_r[k] = \{x \in \{0, 1\}^{m \times m} | m \geq k \& (\text{there exist exactly } k \text{ 1s both on the first and on the second rows of } x) \& (\text{the symbol (on the second row) just below the } r\text{th 1 (counting from left to right) on the first row of } x \text{ is } 1)\}$. It is easily seen that for each $k \geq 1$ and for each $r(1 \leq r \leq k)$, $T_r[k] \subseteq \mathcal{L}[TDF]$. From this and the fact that $T[k] = T_1[k] \cap \cdots \cap T_k[k]$, it follows that part (1) of the lemma holds.

(2) Below, we show that $T[k + 1] \subseteq \mathcal{L}[TAT(L(m), k)]$ for any $k \geq 1$ and any $L$ such that $L(ra) \sim \log ra$. From this and Lemma 3.1, it follows that part (2) holds.

Suppose that there is a TAT($L(m), k$) $M$ (with $L(m) \leq \log m$) accepting $T[k + 1]$. We assume without loss of generality that $M$ enters an accepting state only on the bottom boundary symbol #. Let $s$ and $r$ be the numbers of states (of the finite control) and of storage tape symbols of $M$, respectively. For each accepting computation tree $t$ of $M$, let $SC(t)$ be a “multi-set” of semi-configurations of $M$ defined as follows (see Definition 2.3 for semi-configurations):

$$SC(t) = \{(2, j), (q, a, i)\}$$

where $c = (x, ((2, j), (q, a, i)))$ is a node label of $t$, and $c$ is a configuration of $M$ just after the input head reached the second row of $x$, where $x$ is the input associated with $t$. For each input $x$, let $ACT(x)$ be the set of all accepting computation trees of $M$ on $x$ whose leaf-sizes are at most $k$. Furthermore, for each $m \geq k + 1$, let

$$V(m) = \{x \in T[k + 1] | l_1(x) = l_2(x) = m \& x((3, 1), (m, m)) \subseteq \{0\}^{(2)}\}$$

and for each $x$ in $V(m)$ let $C(x) = \{SC(t) | t \in ACT(x)\}$. (Clearly, each tape $x$ in $V(m)$ is accepted by $M$, and so it follows, since we assumed that $M$ enters an accepting state only on the bottom boundary symbol #, that for each $x$ in $V(m)$ $C(x)$ is not empty.) Then, it is easily seen that for any two different tapes $x, y \in V(m)$, $C(x) \cap C(y) = \emptyset$.

Let $p(m)$ be the number of possible semi-configurations of $M$ just after the input head reached the second rows of tapes in $V(m)$. Then

$$p(m) \leq s(m + 2) L(m) r^{L(m)}.$$ 

Since for each $x$ in $V(m)$ and for each $t$ in $ACT(x)$ $LEAF(t)$ is at most $k$, it follows that for each $x$ in $V(m)$ and for each $t$ in $ACT(x)$ $|SC(t)| \leq k$.

$^7$ For any set $A$, $|A|$ denotes the number of elements of $A$. 
Therefore, letting $S(m) = \{|SC(t)| t \in ACT(x) \text{ for some } x \in V(m)\}$, it follows that for some constants $c$ and $c'$,

$$|S(m)| \leq c p(m)^k \leq c' m^k L(m)^k s^{kL(m)}.$$ 

As is easily seen, $V(m) = (k^m)$. Since $L(m) \leq \log m$, we have $|S(m)| \leq |V(m)|$ for large $m$. Therefore for large $m$ there must be different tapes $x, y$ in $V(m)$ such that $C(x) \cap C(y) \neq \emptyset$. This is a contradiction. Q.E.D.

From Lemma 3.2, we can get

**Theorem 3.1.** For any integer $k \geq 1$ and for any $L : N \to R$ such that $L(m) \leq \log m$,

1. $L_k[TNT(L(m))] = L_{k+1}[TNT(L(m))]$, and
2. $L_k[TDT(L(m))] = L_{k+1}[TDT(L(m))]$.

**Corollary 3.1.** For any integer $k \geq 1$,

1. $L_k[TNF] = L_{k+1}[TNF]$, and
2. $L_k[TDF] = L_{k+1}[TDF]$.

As shown in the next theorem, if $L(m) \geq \log m$, then a situation which differs from Theorem 3.1 emerges.

**Fact 3.1 (Inoue and Takanami, 1979).** $L[TNT(L(m))]$ and $L[TDT(L(m))]$ are closed under finite intersection for any $L$ such that $L(m) \geq \log m (m \geq 1)$.

**Theorem 3.2.** For any integer $k \geq 1$ and for any function $L(m) \geq \log m (m \geq 1)$,

1. $L_k[TNT(L(m))] = L_1[TNT(L(m))] = L[TNT(L(m))]$, and
2. $L_k[TDT(L(m))] = L_1[TDT(L(m))] = L[TDT(L(m))]$.

**Proof.** The theorem follows from Fact 3.1. Q.E.D.

**Remark 3.1.** It is easily seen that $L[NT(L(m))]$ and $L[DT(L(m))]$ are closed under intersection for any $L$. From this fact, it follows that $L_k[NT(L(m))] = L[NT(L(m))]$ and $L_k[DT(L(m))] = L[DT(L(m))]$ for any integer $k \geq 1$ and any $L$.

We next derive the main theorem of this section. We need the following two lemmas.

**Lemma 3.3.** $L[TNT(L(m))] \subseteq \bigcup_{c>0} L[TDT(mL(m)c^{L(m)})]$ for any function $L : N \to R$. 

Proof. Let $M$ be a TNT($L(m)$), and $s$ and $r$ be the numbers of states (of the finite control) and storage tape symbols of $M$, respectively. We construct a TDT($mL(m)\cdot L(m)^r$) $M'$ accepting the set $T(M)$, where $mL(m)\cdot L(m)^r > (m+2)sL(m)\cdot rL(m)(m \geq 1)$. Suppose that an input tape $x$ with $l_1(x) = l_2(x) = m$ ($m \geq 1$) is presented to $M'$. Let $C$ be the set of possible storage states of $M$ on $x$. Clearly, $|C| = (m+2)sL(m)\cdot rL(m)$. For each $i(1 \leq i \leq l_1(x) + 1 = m + 1)$, let $H(i)$ be a subset of $\{0, 1, \ldots, m + 1\} \times C$ defined as follows:

$H(1) = \{(1, p_0)\}$, where $p_0$ is the initial storage state of $M$ (i.e., $p_0$ is the storage state component of the initial configuration of $M$ on $x$).

$H(i), i > 1 = \{(j, p) \mid I_M(x) \vdash_M (x, ((i-1, j), p')) \vdash_M (x, ((i, j), p))$ for some $p' \in C$, i.e., there exists a computation path (of $M$ on $x$) leading to the configuration $(x, ((i, j), p))$ such that $(x, ((i, j), p))$ is the configuration just after the input head of $M$ has reached the $i$th row of $x$. $M'$ simulates the action of $M$ on $x$ by successively generating $H(1)$, $H(2)$, $\ldots$, $H(i)$, $\ldots$, on the storage tape, in such a way that for each $i > 1$, $M'$ replaces $H(i)$ with $H(i+1)$ on the storage tape. Note that, since, by assumption, $mL(m)\cdot L(m)^r > |H(i)|$ for each $1 \leq i \leq m + 1$, $M'$ can store each element of $H(i)$ on the storage tape. It will be easy to see that $M'$ can generate $H(i+1)$ from $H(i)$ for each $1 \leq i \leq m$. The details are left to the reader.

$M'$ enters an accepting state if and only if it finds out that for some $i(1 \leq i \leq m + 1)$, $H(i)$ contains at least one element $(j, p)$ such that $0 \leq j \leq m + 1$ and $p$ is a storage state whose state-component is an accepting state of $M$. It will be obvious that $T(M') = T(M)$. Q.E.D.

Lemma 3.4. Let $T_1 = \{x \in \{a, b, c\}^{2m \times 2m} \mid m \geq 2 \& \exists i (1 \leq i \leq m) [x(i + m, 1) = c \& \forall j (2 \leq j \leq 2m) [x(i, j) \in \{a, b\} \& x(i + m, j) \in \{a, b\} \& \forall r (i + m + 1 \leq r \leq 2m) [x(r, 1) \in \{a, b\} \& x[(i, 2), (i, 2m)] \neq x[(i + m, 2), (i + m, 2m)]]]\}$. Then

(1) $T_1 \in \mathcal{L}[\text{TAF}(2)]$;

(2) $T_1 \in \mathcal{L}^k[\text{TNT}(L(m))]$ for any integer $k \geq 1$ and for any $L : N \rightarrow R$ such that $L(m) \ll \log m$.

Proof. (1): The set $T_1$ is accepted by a TAF(2) $M$ which acts as follows. Given an input $x$ ($l_1(x) = l_2(x) = 2m, m \geq 1$), starting on the upper left-hand corner of $x$, $M$ existentially chooses some $i, j(1 \leq i \leq 2m, 2 \leq j \leq 2m)$, to pick up the symbol $x(i, j)$, and to store it in the finite control. Then $M$ enters a universal state to choose one of two further actions:

(a) One action is to check whether the segment $x[(i, 2), (i, 2m)]$ does not contain the symbol $c$. If so, $M$ moves right until it reaches the right boundary symbol #. Then $M$ continues to move its input head $H$ one cell
down for every two left moves of $H$. If $H$ meets the symbol $c$ on the leftmost column, $M$ then checks that the segment (on the leftmost column) below this cell does not contain the symbol $c$. $M$ enters an accepting state, if this check is successful.

(b) The other action is to existentially choose one of the following two actions, each time $H$ meets a symbol which differs from the symbol $x(i,j)$ stored in the finite control:

(i) One action is to continue to move down along the $j$th column, seeking for another symbol different from $x(i,j)$. (In this case, $M$ will not enter an accepting state on the way.)

(ii) The other action is to move $H$ to the left, and to check that $H$ meets the symbol $c$. If this check is successful, then $M$ checks that just the same row does not contain $c$ except the leftmost cell, and that the segment (on the leftmost column) below the leftmost cell does not contain $c$. $M$ enters an accepting state, if $M$ succeeds in these checks.

It will be obvious that $M$ accepts $T_1$.

(2) Suppose that $T_1$ is in $\mathcal{L}^k[TNT(L(m))]$ for some $k \geq 1$, where $L(m) \ll \log m$. Let $T_1 = A_1 \cap \cdots \cap A_k$, where each $A_i \in \mathcal{L}[TNT(L(m))]$. Since $mL(m)h^L(m) \ll m^2$ for any constant $h > 0$, it follows from Lemma 3.3 that each $A_i$ is in $\mathcal{L}[TDT(L'(m))]$ for some function $L'(m)$ such that $L'(m) \geq m (m \geq 1)$ and $L'(m) \ll m^2$. Therefore, from Fact 3.1, it follows that $T_1$ is in $\mathcal{L}[TDT(L'(m))]$. On the other hand, by using the same technique as in the proof of Lemma 3.5 (2) in Inoue and Takanami (1980), we can easily show that $T_1$ is not in $\mathcal{L}[TDT(L'(m))]$, which is a contradiction. Q.E.D.

From Lemma 3.1 and 3.4, we get the following main theorem.

**Theorem 3.3.** For any integer $k \geq 2$ and for any $L : N \rightarrow R$ such that $L(m) \ll \log m$,

$$\mathcal{L}^k[TNT(L(m))] \subsetneq \mathcal{L}[TAT(L(m), k)].$$

**Corollary 3.2.** For any integer $k \geq 2$,

$$\mathcal{L}^k[TNF] \subsetneq \mathcal{L}[TAF(k)].$$

**Remark 3.2.** It is shown (Inoue et al., 1982, 1983) that $\mathcal{L}[TAT(L(m), k)] = \mathcal{L}[TNT(L(m))]$ for any $k \geq 1$ and any $L(m) \geq \log m$. From this and Theorem 3.2, it follows that $\mathcal{L}^k[TNT(L(m))] = \mathcal{L}[TAT(L(m), k)]$ for any $k \geq 1$ and any $L(m) \geq \log m$. 
4. THE ACCEPTING POWER OF TUTS

We denote by TUT (TUF) a TAF (TAF) which has only universal states. For any function \( L: \mathbb{N} \rightarrow \mathbb{R} \), we denote by TUT\( (L(m)) \) an \( L(m) \) space bounded TUT, and let \( \mathcal{L}[\text{TUT}(L(m))] = \{ T \mid T = T(M) \text{ for some TUT}(L(m)) M \} \). \( \mathcal{L}[\text{TUF}] \) is defined in a similar way.

In this section, we investigate the relationship between the accepting power of TUTs and TATs (TNTs or TDTs).

The following lemma says that there exists a set accepted by a TNF, but not accepted by any TUT\( (L(m)) \) for any \( L \) such that \( L(m) \ll m \).

**Lemma 4.1.** Let \( T_2 = \{ x \in \{ a, b \}^{m \times m} \mid m \geq 2 \text{ and } x[1, \ast] = x[2, \ast] \} \). Then

1. \( T_2 \in \mathcal{L}[\text{TNF}] \), and
2. \( T_2 \not\in \mathcal{L}[\text{TUT}(L(m))] \) for any \( L: \mathbb{N} \rightarrow \mathbb{R} \) such that \( L(m) \ll m \).

**Proof.** We only give the proof of (2), since (1) is easy to prove. Suppose that there exists a TUT\( (L(m)) M \) accepting \( T_2 \), where \( L(m) \ll m \). Let \( s \) and \( r \) be the numbers of states (of the finite control) and storage tape symbols of \( M \), respectively. For each \( m \geq 3 \), let

\[
V(m) = \{ x \in \{ a, b \}^{m \times m} \mid x[1, \ast] = x[2, \ast] \text{ and } x[(3, 1), (m, m)] \subseteq \{ a \} \}.
\]

For each \( x \in V(m) \), let \( S(x) \) and \( C(x) \) be sets of semi-configurations of \( M \) defined as follows:

\[
S(x) = \{ ((2, j), (q, a, i)) \mid \text{there exists a computation path of } M \text{ on } x, I_m(x) \preceq_M (x, ((1, j), (q', a', i'))) -_M (x, ((2, j), (q, a, i))) \}.
\]

\[
C(x) = \{ \sigma \in S(x) \mid \text{when, starting with the configuration } (x, \sigma), M \text{ proceeds to read the segment } x[2, \ast] \text{, there exists a sequence of steps of } M \text{ in which } M \text{ never enters an accepting state} \}.
\]

(Note that, for each \( x \in V(m) \), \( C(x) \) is not empty, since \( x \) is not in \( T_2 \), and so not accepted by \( M \).) Then the following proposition must hold.

**Proposition 4.1.** For any two different tapes \( x, y \) in \( V(m) \), \( C(x) \cap C(y) = \emptyset \).

**Proof.** For otherwise, suppose that \( x \neq y \) \( (x, y \in V(m)) \), \( C(x) \cap C(y) \neq \emptyset \), and \( \sigma \in C(x) \cap C(y) \). Consider the tape \( z \) (with \( l_1(z) = l_2(z) = m \)) satisfying the following (i) and (ii).

1. \( z[1, \ast] = x[1, \ast] \);
2. \( z[(2, 1), (m, m)] = y[(2, 1), (m, m)] \).

\( ^8 \) If \( T \subseteq \Sigma^{(2)} \), then define \( \overline{T} = \Sigma^{(2)} - T \).
From (i) and the assumption that $\sigma \in C(x)$, it follows that there exists a computation path of $M$ on $z$, $I_M(z) \leq M(z, \sigma)$. Further, from (ii) and the assumption that $\sigma \in C(y)$, it follows that when, starting with the configuration $(z, \sigma)$, $M$ proceeds to read the segment $z[(2, 1), (m, m)]$, there exists a sequence of steps of $M$ in which $M$ never enters an accepting state. This means that $z$ is not accepted by $M$. (Note that all states of $M$ are universal.) This contradicts the fact that $z$ is in $T_2 = T(M)$. Q.E.D.

Clearly, $|V(m)| = 2^m$, and $p(m) \leq s(m + 2) L(m) r^{L(m)}$, where $p(m)$ denotes the number of possible semi-configurations of $M$ just after the input head reached the second rows of tapes in $V(m)$. Since $L(m) \leq m$, we have $|V(m)| > p(m)$ for large $m$. Therefore, it follows that for large $m$ there must be two different tapes $x, y$ in $V(m)$ such that $C(x) \cap C(y) \neq \emptyset$. This contradicts Proposition 4.1 and completes the proof of (2). Q.E.D.

The following lemma says that there exists a set accepted by a $\text{TNT}(\log m)$, but not accepted by any $\text{TUT}(L(m))$ for any $L$ such that $L(m) \leq m^2$.

**Lemma 4.2.** Let $T_3 = \{x \in \{a, b\}^{2m \times 2m} | m \geq 1 \& x[(1, 1), (m, 2m)] = x[(m + 1, 1), (2m, 2m)]\}$. Then

1. $T_3 \in \mathcal{L}[\text{TNT}(\log m)]$, and
2. $T_3 \notin \mathcal{L}[\text{TUT}(L(m))]$ for any $L : N \rightarrow R$ such that $L(m) \leq m^2$.

**Proof.** The proof of (1) is given in the proof of Theorem 5.3 in Inoue and Takanami (1979). The proof of (2) is quite similar to that of Lemma 4.1(2). So the proof is left to the reader. Q.E.D.

We need the following two lemmas.

**Lemma 4.3.** Let $T_2$ be the set described in Lemma 4.1. Then

1. $T_2 \in \mathcal{L}[\text{TUF}]$, and
2. $T_2 \notin \mathcal{L}[\text{TNT}(L(m))]$ for any $L : N \rightarrow R$ such that $L(m) \leq m^2$.

**Proof.** The proof of (1) is easy. The proof of (2) is given in the proof of Lemma 3.1 (2) in Inoue and Takanami (1979). Q.E.D.

**Lemma 4.4.** Let $T_3$ be the set described in Lemma 4.2. Then

1. $T_3 \in \mathcal{L}[\text{TUT}(\log m)]$, and
2. $T_3 \notin \mathcal{L}[\text{TNT}(L(m))]$ for any $L : N \rightarrow R$ such that $L(m) \leq m^2$.

**Proof.** The set $T_3$ is accepted by a $\text{TUT}(\log m)$ which, given an input $x \in \{a, b\}^{2m \times 2m}$, simply checks by using universal states that for each $i, j$
Theorem 4.1. Let $L : N \rightarrow R$ be a function such that (i) $L(m) \ll m$, or (ii) $L(m) \gg \log m (m \geq 1)$ and $L(m) \ll m^2$. Then

1. $\mathcal{L}[\text{TUT}(L(m))] \nsubseteq \mathcal{L}[\text{TAT}(L(m))]$,
2. $\mathcal{L}[\text{TUT}(L(m))]$ is incomparable with $\mathcal{L}[\text{TNT}(L(m))]$, and
3. $\mathcal{L}[\text{TDT}(L(m))] \nsubseteq \mathcal{L}[\text{TUT}(L(m))]$.

Corollary 4.1. (1) $\mathcal{L}[\text{TUF}] \nsubseteq \mathcal{L}[\text{TAF}]$. (2) $\mathcal{L}[\text{TUF}]$ is incomparable with $\mathcal{L}[\text{TNF}]$. (3) $\mathcal{L}[\text{TDF}] \subseteq \mathcal{L}[\text{TUF}]$.

By using a technique similar to that in the proof of Lemma 3.3, we can show that, given a TUF $M$, one can construct a TDT$(m)$ accepting the set $T(M)$. (The proof is easy.) Thus it follows that $\mathcal{L}[\text{TUF}] \subseteq \mathcal{L}[\text{TDT}(m)]$. On the other hand, as is easily seen, $T_2 \subseteq \mathcal{L}[\text{TDT}(m)]$, where $T_2$ is the set described in Lemma 4.1. From these facts and Lemma 4.3, we can get the following result.

Theorem 4.2. (1) $\mathcal{L}[\text{TUF}] \subseteq \mathcal{L}[\text{TDT}(m)]$. (2) $m$ space is necessary and sufficient for TDTs and TNTs to simulate TUFs.

Remark 4.1. It is shown (Inoue et al., 1982, 1983) that $m^2$ space is necessary and sufficient for TDTs to simulate TAFs.

5. The Accepting Power of UTs

We denote by UT (UF) an AT (AF) which has only universal states. For any function $L : N \rightarrow R$, we denote by UT$(L(m))$ an $L(m)$ space bounded UT, and let $\mathcal{L}[\text{UT}(L(m))] = \{ T \mid T = T(M) \text{ for some UT}(L(m)) M \}$. $\mathcal{L}[\text{UF}]$ is defined in a similar way. This section first investigates a relationship between the accepting powers of UTs and ATs (NTs or DTs).

We first give several preliminaries to get the desired result. Let $\Sigma$ be a finite alphabet. For each $m \geq 2$ and each $1 \leq n \leq m - 1$, an $(m, n)$-chunk over $\Sigma$ is a pattern $x$ over $\Sigma$ as shown in Fig. 2, where $x_1 \in \Sigma^{(2)}$, $x_2 \in \Sigma^{(2)}$, $l_1(x_1) = m - 1$, $l_2(x_1) = n$, $l_1(x_2) = m$, and $l_2(x_2) = m - n$. Let $M$ be a UT$(l)$. Note that if the numbers of states and storage-tape symbols of $M$ are $s$ and $r$, respectively, then the number of possible storage states of $M$ is $slr^l$. Let $\Sigma$ be the input alphabet of $M$, and let $\#$ be the boundary symbol of $M$. For any $(m, n)$-chunk $x$ over $\Sigma$, we denote by $x(\#)$ the pattern (obtained from $x$ by
surrounding $x$ by $\#$'s) as shown in Fig. 3. Below, we assume without loss of generality that for any $(m, n)$-chunk over $\Sigma$ ($m \geq 2$, $1 \leq n \leq m - 1$), $M$ has the property (A)\(^9\):

(A) $M$ enters or exists the pattern $x(\#)$ only at the face designated by the bold line in Fig. 3, and $M$ never enters an accepting state in $x(\#)$.

Then the number of the entrance points to $x(\#)$ (or the exit points from $x(\#)$) for $M$ is $n + 3$. We suppose that these entrance points (or exit points) are numbered $1, 2, \ldots, n + 3$ in an appropriate way. Let $P = \{1, 2, \ldots, n + 3\}$ be the set of these entrance points (or exit points). Let $C = \{q_1, q_2, \ldots, q_u\}$ be the set of possible storage states of $M$, where $u = \text{str}^t$. For each $i \in P$ and each $q \in C$, let $M_{(i, q)}(x(\#))$ be a subset of $P \times C \cup \{L\}$ which is defined as follows ($L$ is a new symbol):

1. $(j, p) \in M_{(i, q)}(x(\#))$ $\iff$ when $M$ enters the pattern $x(\#)$ in storage state $q$ and at point $i$, there exists a sequence of steps of $M$ in which $M$ eventually exists $x(\#)$ in storage state $p$ and at point $j$.

2. $L \in M_{(i, q)}(x(\#))$ $\iff$ when $M$ enters the pattern $x(\#)$ in storage state $q$ and at point $i$, there exists a sequence of steps of $M$ in which $M$ never

\(^9\) Note that for any $UT(l) M'$, we can construct a $UT(l) M$ with the property (A) such that $T(M) = T(M')$. 
exits \( x(\#) \). (Note the assumption that \( M \) never enters an accepting state in \( x(\#) \).)

Let \( x, y \) be any two \((m, n)\)-chunks over \( \Sigma \). We say that \( x \) and \( y \) are \( M \)-equivalent if for each \((i, q) \in P \times C\), \( M_{(i,q)}(x(\#)) = M_{(i,q)}(y(\#)) \). For any \((m, n)\)-chunk \( x \) over \( \Sigma \) and for any tape \( v \in \Sigma^{1 \times n} \), let \( x[v] \) be the tape in \( \Sigma^{(2)} \) consisting of \( v \) and \( x \) as shown in Fig. 4.

The following lemma means that \( M \) cannot distinguish between two \((m, n)\)-chunks which are \( M \)-equivalent.

**Lemma 5.1.** Let \( M \) be a UT(1) with the property (A), and \( \Sigma \) be the input alphabet of \( M \). Let \( x \) and \( y \) be \( M \)-equivalent \((m, n)\)-chunks over \( \Sigma \) \((m \geq 2, 1 \leq n \leq m - 1)\). Then, for any tape \( v \in \Sigma^{1 \times n} \), \( x[v] \) is accepted by \( M \) if and only if \( y[v] \) is accepted by \( M \).

**Proof.** The lemma follows from the observation that there exists an accepting computation tree of \( M \) on \( x[v] \) if and only if there exists an accepting computation tree of \( M \) on \( y[v] \), since \( x \) and \( y \) are \( M \)-equivalent.

Q.E.D.

Clearly, \( M \)-equivalence is an equivalence relation on \((m, n)\)-chunks, and we get

**Lemma 5.2.** Let \( M \) be a UT(1) with the property (A) above, and \( \Sigma \) be the input alphabet of \( M \). Then there are at most

\[
(2^{(n+3)u+1})(n+3)^u
\]

\( M \)-equivalence classes of \((m, n)\)-chunks over \( \Sigma \), where \( u = s \cdot r' \), \( s \) is the number of states of the finite control of \( M \), and \( r \) is the number of storage tape symbols of \( M \).

**Proof.** The proof is similar to that of Lemma 2.1 in Inoue and Takanami (1978).

Q.E.D.

We are now ready to prove the following key lemma.

![Fig. 4. \( x[v] \).](image-url)
Lemma 5.3. Let \( T_4 = \{ x \in \{ a, b \}^{m \times m} | m \geq 2 \ \& \ \exists i (2 \leq i \leq m) | x[i, \ast] = x[1, \ast] \} \). Then

(1) \( T_4 \in \mathcal{L}[AF] \), and

(2) \( T_4 \not\in \mathcal{L}[UT(L(m))] \) for any \( L : N \rightarrow R \) such that \( L(m) \ll \log m \).

Proof. The set \( T_4 \) is accepted by an AF \( M \) which acts as follows: Given an input \( x \) (\( l_1(x) = l_2(x) = m \geq 2 \)), \( M \) existentially (i.e., in existential states) chooses some row, say the \( i \)th row, of \( x \). Then \( M \) universally tries to check that for each \( j (1 \leq j \leq m) \), \( x(i, j) = x(1, j) \). \( M \) enters an accepting state if this check is successful. (The details of the action of \( M \) are left to the reader.) Below, we prove (2). Suppose that there is a \( UT(L(m)) \) \( M' \) accepting \( T_4 \), where \( L(m) \sim \log m \). Let \( s \) and \( r \) be the numbers of states (of the finite control) and storage tape symbols of \( M' \), respectively. We assume without loss of generality that when \( M' \) accepts a tape \( x \) in \( T_4 \), it enters an accepting state only on the upper left-hand corner of \( x \) and that \( M' \) never falls off an input tape out of the boundary symbol \#. (Thus, \( M' \) satisfies the property (A).) For each \( n \geq 1 \), let

\[
V(n) = \{ x \in \{ a, b \}^{(2^n + 1) \times (2^n + 1)} | x[(1, 1), (2^n + 1, n)] \}
\]

\[
\in \{ a, b \}^{(2^n + 2)} \ \& \ x[(1, n + 1), (2^n + 1, 2^n + 1)]
\]

\[
\in \{ a \}^{(2^n)}
\]

\[
Y(n) = \{ y \in \{ a, b \}^{1 \times n} \}
\]

and

\[
R(n) = \{ \text{row}(x) | x \in V(n) \},
\]

where for each \( x \) in \( V(n) \), \( \text{row}(x) = \{ y \in Y(n) | y = x[(i, 1), (i, n)] \} \) for some \( i \), \( 2 \leq i \leq l_1(x) = 2^n + 1 \). Since \( |Y(n)| = 2^n \), it follows that \( |R(n)| \) is the number of nonempty subsets of \( Y(n) = 2^{2n} - 1 \). Note that \( B = \{ p | \text{for some } x \in V(n), p \text{ is the pattern obtained from } x \text{ by cutting the part } x[(1, 1), (1, n) \text{ off} \} \) is the set of all \( (2^n + 1, n) \)-chunks. Since \( M' \) can use at most \( L(2^n + 1) \) cells of the storage tape when \( M' \) reads a tape in \( V(n) \), from Lemma 5.2, it follows that there are at most

\[
E(n) = (2^{(n+3)u[n]+1})^{(n+3)u[n]}
\]

\( M' \)-equivalence classes of \( (2^n + 1, n) \)-chunks, where \( u[n] = sL(2^n + 1) \) \( r^L(2^n + 1) \). We denote these \( M' \)-equivalence classes by \( C_1, C_2, \ldots, C_{E(n)} \). Since \( L(m) \ll \log m \) (by assumption), it follows that \( L(2^n + 1) \ll \log(2^n + 1) \), and so \( L(2^n + 1) \ll n \). By using this fact, we have \( |R(n)| > E(n) \) for large \( n \). For such \( n \), there must be some \( Q, Q'(Q \neq Q') \) in \( R(n) \) and some \( C_i \) \((1 \leq i \leq E(n)) \) such that the following statement holds:
There exist two tapes $x, y$ in $V(n)$ such that

(i) for some row $\rho$ in $Q$ but not in $Q'$, $x[(1, 1), (1, n)] = y[(1, 1), (1, n)] = \rho$,

(ii) row($x$) = $Q$ and row($y$) = $Q'$, and

(iii) both $p_x$ and $p_y$ are in $C_i$, where $p_x(p_y)$ is the $(2^n + 1, n)$-chunk obtained from $x$ (from $y$) by cutting the part $x[(1, 1), (1, n)]$ (the part $y[(1, 1), (1, n)]$) off.

As is easily seen, $x$ is in $T_4$, and so $x$ is accepted by $M'$. Therefore, from Lemma 5.1, it follows that $y$ is also accepted by $M'$, which is a contradiction. (Note that $y$ is not in $T_4$.) This completes the proof of (2).

Q.E.D.

Further, we need

**Lemma 5.4.** Let

$$T_5 = \{x \in \{a, b\}^{(2m+1) \times (2m+1)} | m \geq 0 \& x(m + 1, m + 1) = b \}$$

(that is, the center symbol of $x$ is $b$). Then

(1) $T_5 \subseteq L[TUF] \subseteq L[UF]$, and

(2) $T_5 \not\subseteq L[DT(L(m))]$ for any $L : N \rightarrow R$ such that $L(m) \leq \log m$.

**Proof.** (1) The set $T_5$ is accepted by a TUF $M$ which acts as follows. Suppose that an input $x$ with $l_1(x) = l_2(x) = 2m + 1$ ($m \geq 0$) is presented to $M$. On each symbol $x(i, i)$, $1 \leq i \leq 2m + 1$, $M$ enters a universal state to choose one of two further actions. One action is to simply move to the next symbol $x(i + 1, i + 1)$ by moving along the diagonal. In this case, if $M$ meets the rightmost and lowermost boundary symbol $\#$, then $M$ enters an accepting state. The other action is to pick up and store the symbol $x(i, i)$ in the finite control, to make a $90^\circ$ right turn, and to move toward the border. In this case, if the symbol stored above is "a" and $M$ meets the leftmost and lowermost boundary symbol $\#$, then $M$ never enters an accepting state, and otherwise $M$ enters an accepting state. It will be obvious that $T(M) = T_5$.

(2) The proof of (2) is given in the Appendix. Q.E.D.

From Lemmas 5.3 and 5.4, we get

**Theorem 5.1.** Let $L : N \rightarrow R$ be a function such that $L(m) \leq \log m$. Then, $L[DT(L(m))] \not\subseteq L[UT(L(m))] \subseteq L[AT(L(m))]$.

**Remark 5.1.** We conjecture that $L[NT(L(m))]$ is incomparable with $L[UT(L(m))]$ for any $L$ such that $Lm) \leq \log m$. But we have no proof of this conjecture. It is also unknown whether or not $L[DT(L(m))] \not\subseteq$...
$L[UT(L(m))] \subseteq L[AT(L(m))]$ for any $L(m) \geq \log m$. By using a technique similar to that in the proof of Theorem 4.2 in Chandra et al. (1981), we can show that $L[UT(L(m))] \subseteq L[DT(L(m)^2)]$ for any $L(m) \geq \log m$.

As a corollary of Theorem 5.1, we get

**Corollary 5.1.** $L[DF] \not= L[UF] \not= L[AF]$.

We then investigate relationships between the accepting powers of UTs and three-way Turing machines.

**Lemma 5.5.** Let $T_2$ be the set described in Lemma 4.1. Then

(1) $T_2 \subseteq L[DF]$, and
(2) $T_2 \not\subseteq L[TUT(L(m))]$ for any function $L: N \to R$ such that $L(m) \leq m$.

**Proof.** Part (1) is obvious. Part (2) is given in Lemma 4.1(2). Q.E.D.

**Lemma 5.6.** Let $T_3$ be the set described in Lemma 4.2. Then

(1) $T_3 \subseteq L[DT(\log m)]$, and
(2) $T_3 \not\subseteq L[TUT(L(m))]$ for any $L: N \to R$ such that $L(m) \leq m^2$.

**Proof.** Part (1) is obvious. Part (2) is given in Lemma 4.2(2). Q.E.D.

From Lemmas 5.5 and 5.6, we get

**Theorem 5.2.** Let $L: N \to R$ be a function such that (i) $L(m) \leq m$, or (ii) $L(m) \geq \log m \,(m \geq 1)$ and $L(m) \leq m^2$. Then, $L[TUT(L(m))] \not\subseteq L[UT(L(m))]$.

**Corollary 5.2.** $L[TUF] \not\subseteq L[UF]$.

**Remark 5.2.** From Lemmas 5.4 and 5.5, it follows, for example, that for any $L$ such that $L(m) \leq \log m$,

(1) $L[DT(L(m))]$ and $L[DF]$ are incomparable with $[TUT(L(m))]$, and
(2) $L[TUF]$ is incomparable with $L[DT(L(m))]$.

We conjecture that $L[UT(L(m))]$ is incomparable with $L[TAT(L(m))]$ for any $L$ such that $L(m) \leq \log m$. But we have no proof of this conjecture. (Let $T_6 = \{ x \in \{a, b\}^{m \times m} | m \geq 2 \text{ and } \exists 1 \leq i \leq m - 1 \} \{ x[i, *] = x[m, *] \} \}. \text{ It is shown in Inoue et al., (1982, 1983) that } T_6 \in L[TAF]. \text{ On the other hand, by using the same technique as in the proof of Lemma 5.3(2), we can show that } T_6 \not\subseteq L[UT(L(m))] \text{ for any } L \text{ such that } L(m) \leq \log m.$
We finally give the necessary and sufficient space for TDTs to simulate UF.

**Lemma 5.7.** Let \( T_7 = \{ x \in \{a, b, c\}^{2m \times 2m} | m \geq 1 \} \) and \( \exists i (m + 2 \leq i \leq 2m) [x[i, \ast] \in \{c\}^{(2)} \land \forall i' (1 \leq i' \leq 2m, i' \neq i)] [x[i', \ast] \in \{a, b\}^{(2)}] \land x[i-1, \ast] = x[i-m-1, \ast] \}\).

(1) \( T_7 \in \mathcal{L}[UF] \), and

(2) \( T_7 \notin \mathcal{L}[TDT(L(m))] \) for any \( L : N \rightarrow R \) such that \( L(m) \leq m^2 \).

**Proof.** (1) The set \( T_7 \) is accepted by a UF \( M \) which acts as follows: Given an input \( x \) with \( l_1(x) = l_2(x) = 2m (m \geq 1) \), \( M \) first checks that for one and only one \( i (m + 2 \leq i \leq 2m) \), the \( i \)th row of \( x \) consists of only \( c \)'s, and that other rows do not contain the symbol \( c \). If this check is successful, then \( M \) moves toward the leftmost symbol, \( x(i-m-1, 1) \), of the \((i-m-1)\)th row of \( x \). After that, \( M \) universally (i.e., in universal states) checks that for each \( j (1 \leq j \leq 2m) \), \( x(i-m-1, j) = x(i-1, j) \). \( M \) enters an accepting state only if this check is successful. (The details of the action of \( M \) are left to the reader.) It will be obvious that \( T(M) = T_7 \).

(2) The proof of (2) is quite similar to that of Lemma 3.5(2) in Inoue and Takanami (1978). So the proof is omitted here. Q.E.D.

Let \( M \) be a UF, and \( s \) be the number of states of \( M \). Given an input \( x \) with \( l_1(x) = l_2(x) = m \), the number of possible configurations of \( M \) is \( s(m+2)^2 \), which is bounded by \( cm^2 \) for some constant \( c \). From this, it is easily seen that if the input \( x \) is accepted by \( M \), then there is an accepting computation tree of \( M \) on \( x \) whose computation paths from root to leaves each are of length at most \( cm^2 \). From this observation, it is easily ascertained that we can construct, by using the same idea as in the proof of Theorem 3.2 in Chandra et al. (1981), a DT\((m^2)\) \( M' \) which, given an input with \( l_1(x) = l_2(x) = m \), generates every computation path (of \( M \) on \( x \) of length at most \( cm^2 \)) in a systematic way, and checks whether there is an accepting computation tree of \( M \) on \( x \). This implies that \( \mathcal{L}[UF] \subseteq \mathcal{L}[DT(m^2)] \). In Inoue and Takanami (1979), it is shown that \( \mathcal{L}[DT(m^2)] = \mathcal{L}[TDT(m^2)] \). Therefore \( \mathcal{L}[UF] \subseteq \mathcal{L}[TDT(m^2)] \).

Let \( T_4 \) be the set described in Lemma 5.3. From Lemma 5.3(2), it follows that \( T_4 \notin \mathcal{L}[UF] \). On the other hand, as is easily seen, \( T_4 \in \mathcal{L}[TDT(m^2)] \). From these observations and Lemma 5.7, we get

**Theorem 5.3.** (1) \( \mathcal{L}[UF] \nsubseteq \mathcal{L}[TDT(m^2)] \), and (2) \( m^2 \) space is necessary and sufficient for TDTs to simulate UF.

**Remark 5.3.** We conjecture that \( \mathcal{L}[UF] \nsubseteq \mathcal{L}[TNT(m)] \), but we have no proof of this conjecture.
6. Closure Properties of TUTs

This section investigates closure properties of TUTs under Boolean operations. In the one-dimensional case, it is an important open problem (Chandra et al., 1981; Hopcroft and Ullman, 1979) whether or not $\text{co-NPTIME}$ is closed under complementation. We cannot solve this problem, but we show here that $\mathcal{L}[\text{TUT}(L(m))]$ is not closed under complementation for any $L$ such that $L(m) \leq m^2$.

**Theorem 6.1.** Let $L : \mathbb{N} \rightarrow \mathbb{R}$ be a function such that (i) $L(m) \leq m$, or (ii) $L(m) \geq \log m$ ($m \geq 1$) and $L(m) \leq m^2$. Then $\mathcal{L}[\text{TUT}(L(m))]$ is not closed under complementation.

**Proof.** The theorem follows from Lemmas 4.1(2), 4.2(2), 4.3(1), and 4.4(1). Q.E.D.

Next we investigate closure properties of leaf-size bounded TUTs under complementation. Let $L : \mathbb{N} \rightarrow \mathbb{R}$ and $Z : \mathbb{N} \rightarrow \mathbb{R}$ be any function. We denote by $\text{TUT}(L(m), Z(m))$ (TUF($Z(m)$)) a $Z(m)$ leaf-size bounded TUT($L(m)$) (TUF), and let

$$\mathcal{L}[\text{TUT}(L(m), Z(m))] = \{T \mid T = T(m) \text{ for some } \text{TUT}(L(m), Z(m)) M\}.$$  

$\mathcal{L}[\text{TUF}(Z(m))]$ is defined in a similar way. Let $T[k]$ be the set defined in Lemma 3.2.

**Theorem 6.2.** Let $L : \mathbb{N} \rightarrow \mathbb{R}$ be a function such that $L(m) \leq \log m$. Then, for any $Z : \mathbb{N} \rightarrow \mathbb{R}$ such that $Z(m) \geq 2$ ($m \geq 1$), $\mathcal{L}[\text{TUT}(L(m), Z(m))]$ is not closed under complementation.

**Proof.** It is obvious that $T[2] \subseteq \mathcal{L}[\text{TUF}(2)] = \mathcal{L}[\text{TUT}(0, 2)]$. Therefore, in order to prove the theorem, it is sufficient to show that $T[2] \notin \mathcal{L}[\text{TUT}(L(m))]$ for any $L$ such that $L(m) \leq \log m$. To show this, suppose that there exists a TUT($L(m)$) $M$ accepting $T[2]$, where $L(m) \leq \log m$. Let $s$ and $r$ be the numbers of states (of the finite control) and storage tape symbols of $M$, respectively. For each $m \geq 3$, let

$$V(m) = \{x \in T[2] \mid l_1(x) = l_2(x) = m \& x[(3, 1), (m, m)] \in \{0\}^{(2)}\}.$$  

For each $x$ in $V(m)$, let $S(x)$ and $C(x)$ be sets of semi-configurations of $M$ which are defined in the same way as $S(x)$ and $C(x)$, respectively, in the proof of Lemma 4.1(2). Then, by using the same technique as in the proof of Proposition 4.1, we can show that the following proposition must hold.

---

10 See Introduction.
PROPOSITION 6.1. For any two different tapes \( x, y \) in \( V(m) \), \( C(x) \cap C(y) = \emptyset \).

Clearly, \( |V(m)| = m^2 \), and \( p(m) \leq s(m + 2) L(m)^{p(m)} \), where \( p(m) \) denotes the number of possible semi-configurations of \( M \) just after the input head reached the second rows of tapes in \( V(m) \). Since \( L(m) \leq \log m \), we have \( |V(m)| > p(m) \) for large \( m \). Therefore, it follows that for large \( m \) there must be two different tapes \( x, y \) in \( V(m) \) such that \( C(x) \cap C(y) \neq \emptyset \). This contradicts Proposition 6.1. Thus, it follows that \( T_2 \not\subseteq S_2(T(L(m))) \). This completes the proof of the theorem. Q.E.D.

COROLLARY 6.1. For each integer \( k \geq 2 \), \( \mathcal{L}[TUF(k)] \) is not closed under complementation.

In Theorem 4.2 in Inoue et al. (1982, 1983), it is shown that \( \mathcal{L}[TAT(L(m), k)] = \mathcal{L}[TNT(L(m))] \) for any \( L(m) \geq \log m \) and for any integer \( k \geq 1 \). Similarly, we can easily show that \( \mathcal{L}[TUT(L(m), k)] = \mathcal{L}[TDT(L(m))] \) for any \( L(m) \geq \log m \) and for any integer \( k \geq 1 \). On the other hand, it is shown (Inoue and Takanami, 1979) that \( \mathcal{L}[TDT(L(m))] \) is closed under each of Boolean operations for any \( L(m) \geq \log m \). Therefore, we get

\[ \text{THEOREM 6.3. For any function } L(m) \geq \log m \quad (m \geq 1) \quad \text{and for any integer } k \geq 1, \quad \mathcal{L}[TUT(L(m), k)] \text{ is closed under complementation, union, and intersection.} \]

We need the following two definitions for the next theorem.

**DEFINITION 6.1.** A function \( E : N \to R \) is log-space countable if there is a one-dimensional deterministic Turing machine \( M \) which, when given a string of length \( m \), halts after its read-write head has written down the \( k \)-adic notation of the number \( |E(m)| \), for some \( k \geq 2 \), by using at most \( \lfloor \log m \rfloor + 1 \) cells of the storage tape, where \( M \) has a read-only input tape with end markers and one semi-infinite storage tape (Hopcroft and Ullman, 1979).

**DEFINITION 6.2.** Let \( x \) be a two-dimensional tape with \( l_1(x) = l_2(x) = m \). As shown in Fig. 5a, let each tape cell of \( x \) be numbered \( 1, 2, \ldots, m^2 \) from top to bottom and from left to right on the same row. Then, for each \( 1 \leq i \leq j \leq m \), let \( x_{i,j} \) be the segment of \( x \) enclosed by the heavy solid line as shown in Fig. 5b.

\[ \text{THEOREM 6.4. Let } L : N \to R \text{ and } Z : N \to R \text{ be any functions such that} \]

\begin{enumerate}
  \item \( L(m) \geq \log m \quad (m \geq 1) \),
  \item \( L \) is fully space constructible,
\end{enumerate}
(iii) \( Z \) is log-space countable,
(iv) \( |L(m)| |Z(m)| \leq m^2/2 \) \((m \geq 1)\), and
(v) \( 1 \leq Z(m) \).

Then, \( \mathcal{L}[\text{TUT}(L(m), Z(m))] \) is not closed under complementation.

Proof. Let \( T[L, Z] \) be the following set, which depends on the functions \( L \) and \( Z \) in the theorem,

\[
T[L, Z] = \{ x \in \{0, 1\}^{2m \times 2m} \mid m \geq 1 \\
&\text{&} x \ll 1, [L(2m)]|Z(2m)| \gg = x\ll 2m^2 + 1, 2 \\
&\text{&} \quad m^2 + [L(2m)]|Z(2m)| \gg \}.
\]

(Note that, from condition (iv) in the theorem, this set is well defined.) The set \( T[L, Z] \) is accepted by a \( \text{TUT}(L(m), Z(m)) \) \( M \) which acts as follows. Suppose that an input \( x \) with \( l_1(x) = l_2(x) = 2m \) \((m \geq 1)\) is presented to \( M \). While moving on the first row of \( x \), \( M \) first marks off exactly \( |L(2m)| \) cells of the storage tape by using the number \( 2m \) of columns. While again moving on the first row, \( M \) then writes down the \( k \)-adic notation (for some \( k \geq 2 \)) of the number \( |Z(2m)| \) on one track of the storage tape by using the number \( 2m \) of columns. (These actions are possible because of conditions (i)–(iii) in the theorem.) After that, \( M \) universally tries to check that, for each \( 1 \leq i \leq |Z(2m)| \), \( x\ll (i - 1)|L(2m)| + 1, i|L(2m)| \gg = x\ll 2m^2 + (i - 1)|L(2m)| + 1, \\
2m^2 + i|L(2m)| \gg \). The remaining proof is left to the reader as an exercise. To complete the proof, we now show that \( \overline{T[L, Z]} \notin \mathcal{L}[\text{TUT}(L(m), Z(m))] \), where \( L \) and \( Z \) are the functions described in the theorem. The proof technique is similar to that of Lemma 4.1(2). Suppose that there exists a \( \text{TUT}(L(m), Z(m)) \) \( M' \) accepting \( \overline{T[L, Z]} \). Let \( s \) and \( r \) be the numbers of states (of the finite control) and storage tape symbols of \( M' \), respectively. For each \( m \geq 1 \), let

FIG. 5. (a) The numbering of tape cells of \( x \), (b) illustration of \( x\ll i, j \gg \).
$$V(m) = \{ x \in \{0, 1\}^{2m \times 2m} \mid x \langle 1, [L(2m)] [Z(2m)] \rangle$$
$$= x \langle 2m^2 + 1, 2m^2 + [L(2m)] [Z(2m)] \rangle$$
& (both \(x \langle [L(2m)] [Z(2m)] + 1, 2m^2 \rangle\) and \(x \langle 2m^2 + [L(2m)] [Z(2m)] + 1, 4m^2 \rangle\) consist of only 0's).

For each \(x\) in \(V(m)\), let \(S(x)\) and \(C(x)\) be sets of semi-configurations of \(M\) defined as follows:

\[S(x) = \{(2m^2 + 1, j), (q, a, i)\} \mid \text{there exists a computation path of } M' \text{ on } x, I_M(x) \vdash_{M'} (x, ((2m^2 + 1, j), (q, a, i))) \text{ (that is, } (x, ((2m^2 + 1, j), (q, a, i))) \text{ is a configuration of } M' \text{ just after the input head reached the first row of the bottom half of } x)\},\]

\[C(x) = \{ \sigma \in S(x) \} \mid \text{when, starting with the configuration } (x, \sigma), M' \text{ proceeds to read the bottom half of } x, \text{ there exists a sequence of steps of } M' \text{ in which } M' \text{ never enters an accepting state}\}.

(Note that, for each \(x\) in \(V(m)\), \(C(x)\) is not empty, since \(x\) is not in \(T[L, Z]\), and so not accepted by \(M'\).) By using the same technique as in the proof of Proposition 4.1, we can easily show that the following proposition must hold.

**Proposition 6.2.** For any two different tapes \(x, y\) in \(V(m)\), \(C(x) \cap C(y) = \emptyset\).

As is easily seen, \(|V(m)| = 2^{[L(2m)] [Z(2m)]}\), and \(p(m) \leq s(2m + 2) L(2m) r^{L(2m)}\), where \(p(m)\) denotes the number of possible semi-configurations of \(M'\) just after the input head reached the first rows of the bottom halves of tapes in \(V(m)\). From conditions (i) and (v) in the theorem, it follows that \(|V(m)| > p(m)\) for large \(m\). Therefore, it follows that for large \(m\) there must be two different tapes \(x, y\) in \(V(m)\) such that \(C(x) \cap C(y) \neq \emptyset\). This contradicts Proposition 6.2. Thus, \(T[L, Z] \notin \varrho[TUT(L(m), Z(m))]\). This completes the proof of the theorem.

Q.E.D.

We then investigate closure properties of TUTs under union.

**Lemma 6.1.** Let \(T_L = \{ x \in \{0, 1\}^{m \times m} \mid m \geq 2 \& \text{(there exist exactly two 1's on the first row of } x) \& \text{(the symbol (on the second row) just below the left 1 on the first row of } x \text{ is 0)}\}\) and \(T_R = \{ x \in \{0, 1\}^{m \times m} \mid m \geq 2 \& \text{(there exist exactly two 1s on the first row of } x) \& \text{(the symbol (on the second row) just below the right 1 on the first row of } x \text{ is 0)}\}\). Then
(1) $T_i, T_r \in \mathcal{L}[TDF] = \mathcal{L}[TUT(0, 1)]$, and
(2) $T_i \cup T_r \in \mathcal{L}[TUT(L(m))]$ for any $L : N \rightarrow R$ such that $L(m) \leq \log m$.

Theorem 6.5. Let $L : N \rightarrow R$ be a function such that $L(m) \leq \log m$. Then, (1) $\mathcal{L}[TUT(L(m))]$ is not closed under union, and (2) $\mathcal{L}[TUT(L(m), Z(m))]$ is not closed under union for any $Z : N \rightarrow R$ such that $Z(m) \geq 1 (m \geq 1)$.

Corollary 6.2. For each $k \geq 1$, $\mathcal{L}[TUF(k)]$ is not closed under union.

It is unknown whether or not $\mathcal{L}[TUT(L(m))]$ is closed under union for any $L$ such that $L(m) \geq \log m$ and $L(m) \leq m^2$.

We finally investigate closure properties of TUTs under intersection.

Theorem 6.6. For any $L : N \rightarrow R$, $\mathcal{L}[TUT(L(m))]$ is closed under intersection.
Proof. The proof is left to the reader as an easy exercise. Q.E.D.

**Lemma 6.2.** Let $T_r[k]$ be the set defined in the proof of Lemma 3.2(1). Then, for each integer $k \geq 1$ and each $1 \leq r \leq k + 1$,

1. $T_r[k + 1] \in \mathcal{P}[TDF] = \mathcal{P}[TUT(0, 1)]$ and
2. $\bigcap_{s=1}^{k+1} T_s[k+1] \notin \mathcal{P}[TUT(L(m), k)]$ for any $L : N \to R$ such that $L(m) \leq \log m$.

Proof. The proof of (1) is omitted here. To prove (2), we first note that $T[k + 1] = \bigcap_{s=1}^{k+1} T_s[k + 1]$, where $T[k + 1]$ is the set defined before Lemma 3.2. It is shown in the proof of Lemma 3.2(2) that $T[k + 1] \notin \mathcal{P}[TAT(L(m), k)]$ for any $L$ such that $L(m) \leq \log m$. Part (2) of the lemma directly follows from this observation. Q.E.D.

The following theorem follows from Lemma 6.2.

**Theorem 6.7.** Let $L : N \to R$ be a function such that $L(m) \leq \log m$. Then, for each $k \geq 1$, $\mathcal{P}[TUT(L(m), k)]$ is not closed under intersection.

It is unknown whether or not $[TUT(L(m), Z(m))]$ is closed under intersection for any $L$ and $Z$ such that $L(m) \geq \log m$ and $1 \leq Z(m)$.

**Appendix: The Proof of Lemma 5.4(2)**

For each $n \geq 1$, let $U(n) = \{x \in \{a, b\}^{n \times n}\}$. Then any tape in $U(n)$ is called an $n$-chunk. Let $M$ be a DT($l$). Note that if the numbers of states and storage tape symbols of $M$ are $s$ and $r$, respectively, then the number of possible storage states of $M$ is $srl^l$. Let $\{a, b\}$ be the input alphabet of $M$. The number of the entrance points to an $n$-chunk $x$ (or the exit points from $x$) for $M$ is $4n$. We suppose that these entrance points (or exit points) are numbered $1, 2, ..., 4n$ in an appropriate way. Let $P = \{1, 2, ..., 4n\}$ be the set of these entrance points (or exit points). Let $C = \{q_1, q_2, ..., q_u\}$ be the set of possible storage states of $M$, where $u = srl^l$. For each $x \in U(n)$, let $M_x : P \times C \to P \times C \cup \{L\}$ be a mapping which is defined as follows ($L$ is a new symbol):

1. $M_x(i, q) = (j, p)$ if and only if when $M$ enters the $n$-chunk $x$ in storage state $q$ and at point $i$, it eventually exits $x$ in storage state $p$ and at point $j$.
2. $M_x(i, q) = L$ if and only if when $M$ enters the $n$-chunk $x$ in storage state $q$ and at point $i$, it does not exit $x$ at all.

Let $x, y$ be any two $n$-chunks. We say that $x$ and $y$ are $M$-equivalent if for
any \((i, q) \in P \times C, M_x(i, q) = M_y(i, q)\). Thus, \(M\) cannot distinguish between two \(n\)-chunks which are \(M\)-equivalent. Clearly, \(M\)-equivalence is an equivalence relation on \(n\)-chunks, and we easily see that the following proposition holds.

**PROPOSITION A.** Let \(M\) be a DT(I), whose input alphabet is \(\{a, b\}\). There are at most

\[
(4nu + 1)^{4^{nu}}
\]

\(M\)-equivalence classes of \(n\)-chunks, where \(u = STR\), \(S\) is the number of states of the finite control of \(M\), and \(r\) is the number of storage tape symbols of \(M\).

We are now ready for the 

**Proof of Lemma 5.4(2).** The proof is similar to that of Theorem 1 in Blum and Hewitt (1967). Suppose that there is a DT\((L(m))\) accepting \(T_s\), where \(L(m) \ll \log m\). Let \(s\) and \(r\) be the numbers of states (of the finite control) and storage tape symbols of \(M\), respectively. We assume without loss of generality that when \(M\) accepts an input \(x\) in \(T_s\), it halts on the upper left-hand corner of \(x\). For each \(n \geq 0\), let

\[
V(n) = \{x \in \{a, b\}^{2n+1}\}.
\]

Since \(M\) can use at most \(L(2n + 1)\) cells of the storage tape when \(M\) reads a tape in \(V(n)\), from Proposition A, there are at most

\[
E(n) = (4nu[n] + 1)^{4^{nu[n]}}
\]

\(M\)-equivalence classes of \(n\)-chunks, where \(u[n] = sL(2n + 1) r^{L(2n + 1)}\). We denote these \(M\)-equivalence classes by \(C_1, C_2, \ldots, C_{E(n)}\). Clearly, the total number of \(n\)-chunks is \(2^{n^2}\). Since \(L(m) \ll \log m\) (by assumption), \(L(2n + 1) \ll \log(2n + 1)\), and so \(L(2n + 1) \ll \log n\). By using this fact, it follows that for large \(n\), \(2^{n^2} > E(n)\). For such \(n\), there must be some \(C_i\) \((1 \leq i \leq E(n))\) such that \(|C_i| \geq 2\). Let \(x, y\) be two different \(n\)-chunks in \(C_i\) and suppose without loss of generality that for some \(r_1, r_2\) \((1 \leq r_1, r_2 \leq n)\), \(x(r_1, r_2) = b\) and \(y(r_1, r_2) = a\). We consider the two tapes \(z, z' \in \{a, b\}^{(2)}\) satisfying the following three conditions:

(i) \(l_1(z) = l_1(z') = l_2(z') = 2n + 1;\)

(ii) \(z[(n - r_1 + 2, n - r_2 + 2), (2n - r_1 + 1, 2n - r_2 + 1)] = x\) and \(z'[(n - r_1 + 2, n - r_2 + 2), (2n - r_1 + 1, 2n - r_2 + 1)] = y;\)

(iii) \(z\) and \(z'\) are equal except the segment described in (ii).
As is easily seen, $z$ is in $T_5$, and so $z$ is accepted by $M$. Since $x$ and $y$ are $M$-equivalent, it follows that $z'$ is also accepted by $M$, which is a contradiction. (Note that $z'$ is not in $T_5$.) This completes the proof of the Lemma 5.4(2).

Q.E.D.

REFERENCES


