IFIG
RESEARCH
REPORT

INSTITUT FÜR INFORMATIK

REFINING NONDETERMINISM BELOW LINEAR-TIME

Martin Kutrib

IFIG RESEARCH REPORT 0104
JUNE 2001

Institut für Informatik
JLU Gießen
Arndtstraße 2
D-35392 Giessen, Germany
Tel: +49-641-99-32141
Fax: +49-641-99-32149
mail@informatik.uni-giessen.de
www.informatik.uni-giessen.de

JUSTUS-LIEBIG-
UNIVERSITÄT
GIESSEN

IFIG Research Report
IFIG Research Report 0104, June 2001

Refining Nondeterminism Below Linear-Time

Martin Kutrib[1]

Institute of Informatics, University of Giessen

Arndtstr. 2, D-35392 Giessen, Germany

Abstract. Multitape Turing machines with a restricted number of nondeterministic steps are investigated. Fischer and Kintala showed an infinite nondeterministic hierarchy of properly included real-time languages between the deterministic languages and the log-bounded nondeterministic languages. This result is extended to time complexities in the range between real-time and linear-time, and is generalized to arbitrary dimensions.
For fixed amounts of nondeterminism infinite proper dimension hierarchies are presented. The hierarchy results are established by counting arguments. For an equivalence relation and a family of witness languages the number of induced equivalence classes is compared to the number of equivalence classes distinguishable by the model in question. By contradiction the properness of the inclusions is proved.

CR Subject Classification (1998): F.1

[1]E-mail: kutrib@informatik.uni-giessen.de

1 Introduction

One of the central questions in complexity theory asks for the power of nondeterminism in bounded-resource computations. Traditionally, nondeterministic devices have been viewed as having as many nondeterministic guesses as time steps. The studies of this concept of unlimited nondeterminism led, for example, to the famous open LBA-problem or the unsolved question whether or not P equals NP. In order to gain further understanding of the nature of nondeterminism in [4, 11] it has been viewed as an additional limited resource at the disposal of time or space bounded computations.

Motivated by the search for problems that are neither in P nor NP-complete the same authors investigated the so-called β-hierarchy [12]. They considered languages acceptable by polynomial-time bounded Turing machines that make a polylogarithmic number of nondeterministic steps. The hierarchy relies on the exponent of the logarithm. For every $c \in \mathbb{N}$ the class of languages acceptable with $O(\log^c)$ nondeterministic steps is included in the class acceptable with $O(\log^{c+1})$ nondeterministic steps. Clearly, since the hierarchy is in between P and NP none of the inclusions is known to be strict, but on the lower end of the hierarchy they proved P to be the class of languages acceptable with $O(\log)$ nondeterministic transitions.

In [2] limited nondeterminism is added to deterministic complexity classes independent of the computational model for the class. For these Guess-and-Check models the nondeterministically chosen bits are added to the input. If one of the choices can be verified as a witness, then the original input is accepted.

Extensive investigations are also done on limited nondeterminism in the context of finite automata and pushdown automata. In [13] the nondeterminism is restricted dependent on the size of finite automata. The authors prove an infinite nondeterministic hierarchy below a logarithmic bound, and relate the amount of nondeterminism to the number of states necessary for deterministic finite automata to accept the same language.

An automata independent quantification of the inherent nondeterminism in regular languages is dealt with in [6]. The relation between the degree of nondeterminism and the degree of ambiguity is considered in [7]. Recently, measures of nondeterminism in finite automata have been investigated in [10].

Two measures for the nondeterminism in pushdown automata are proposed in [19]. By bounding the number of nondeterministic steps dependent on the length of the input a hierarchy of three classes is obtained. A modification of the measure can be found in [17]. The second measure depends on the depth of the directed acyclic graph that represents a given pushdown automaton. The corresponding proof of an infinite nondeterministic hierarchy of properly included classes is completed in [18]. Recently, a further modification and generalization has been investigated in [9].

A good survey of limited nondeterminism reflecting the state-of-the-art at its time is [5].

Back to our first reference [4] which is part of the pioneering works in this field.

In the early days of complexity theory the proper inclusion between the deterministic and nondeterministic real-time multitape Turing machine languages has been shown [8]. In [4] this result is refined by showing an infinite hierarchy between the deterministic real-time Turing machine languages and the languages acceptable by real-time Turing machines whose number of nondeterministic steps is logarithmically bounded. Observe, that due to the previously mentioned result for the lower end of the β-hierarchy a corresponding hierarchy for polynomial-time computations does not exist. Here we are going to generalize this result to arbitrary dimensions and extend it to time complexities in the range between real-time and linear-time. Recently, in [14] an infinite time hierarchy of deterministic Turing machines in that range has been shown. The hierarchy relies on time bounds of the form $id + r$ where r is a sublinear function. In the present work we exhibit infinite nondeterministic hierarchies of properly included classes for those time bounds. Since the models in question are too weak for diagonalization we use counting arguments that are based on a generalization of a well-known equivalence relation. By calculation different numbers for induced and distinguishable equivalence classes for a family of witness languages the properness of the inclusions are established by contradiction.

A variant of the witness languages and the same technique is used to make another step towards the exploration of the world below linear-time. We present a two-dimensional proper hierarchy where each line is an infinite dimension hierarchy for a fixed amount of nondeterminism, and each column is an infinite nondeterministic hierarchy for a fixed dimension.

In the next section we recall briefly the basic definitions of the model in question and its constructible functions. The central equivalence relation, an upper bound of distinguishable equivalence classes for Turing machines and a preliminary result of a technical flavor are shown. In Section 3 we present the nondeterministic hierarchies for time complexities of the form $id + r$. Finally, in Section 4 the two-dimensional hierarchy is established.

2 Preliminaries

We denote the rational numbers by $\mathbb{Q}$, the integers by $\mathbb{Z}$, the positive integers $\{1, 2, ...\}$ by $\mathbb{N}$ and the set $\mathbb{N} \cup \{0\}$ by $\mathbb{N}_0$. The reversal of a word w is denoted by w^R. For the length of w we write $|w|$. We use $\subseteq$ for inclusions and $\subset$ if the inclusion is strict. Let $e_i = (0, \dots, 0, 1, 0, \dots, 0)$ (the 1 is at position i) denote the ith d-dimensional unit vector, then we define

$$E_d = \{0\} \cup \{e_i \mid 1 \le i \le d\} \cup \{-e_i \mid 1 \le i \le d\}.$$

For a function $f : \mathbb{N}_0 \to \mathbb{N}$ we denote its i-fold composition by $f^{[i]}$, $i \in \mathbb{N}$. If f is increasing then its inverse is defined according to

$$f^{-1}(n) = \min\{m \in \mathbb{N} \mid f(m) \ge n\}.$$

The identity function $n \mapsto n$ is denoted by id. As usual we define the set of functions that grow strictly less than f by

$$o(f) = \{g : \mathbb{N}_0 \to \mathbb{N} \mid \lim_{n\to\infty} \frac{g(n)}{f(n)} = 0\}.$$

In terms of orders of magnitude f is an upper bound of the set

$$O(f) = \{g : \mathbb{N}_0 \to \mathbb{N} \mid \exists\, n_0, c \in \mathbb{N} : \forall\, n \geq n_0 : g(n) \leq c \cdot f(n)\}.$$

Conversely, f is a lower bound of the set $\Omega(f) = \{g : \mathbb{N}_0 \to \mathbb{N} \mid f \in O(g)\}$.

A nondeterministic d-dimensional Turing machine with $k \in \mathbb{N}$ tapes consists of a finite-state control, a read-only one-dimensional one-way input tape and k infinite d-dimensional working tapes. On each tape a read-write head is positioned. At the outset of a computation the Turing machine is in the designated initial state and the input is the inscription of the input tape, all the other tapes are blank. The read-write head of the input tape scans the leftmost symbol of the input whereas all the other heads are positioned on arbitrary tape cells. Dependent on the current state and the currently scanned symbols on the $k+1$ tapes, the Turing machine nondeterministically changes its state, rewrites the symbols at the head positions of the working tapes and possibly moves the heads independently to a neighboring cell. The head of the input tape may only be moved to the right. With an eye towards language recognition the machines have no extra output tape but the states are partitioned in accepting and rejecting states. More formally:

Definition 1 *A* nondeterministic d-dimensional Turing machine *with $k \in \mathbb{N}$ tapes* (NTM_k^d) *is a system* $\langle S, T, A, \delta, s_0, F\rangle$, *where*

1. *S is the finite set of* internal states,
2. *T is the finite set of* tape symbols *containing the* blank *symbol* $\sqcup$,
3. *$A \subseteq T \setminus \{\sqcup\}$ is the set of* input symbols,
4. *$s_0 \in S$ is the* initial state,
5. *$F \subseteq S$ is the set of* accepting states,
6. *δ is the* partial transition function *mapping from $S \times (A \cup \{\sqcup\}) \times T^k$ into the subsets of $S \times T^k \times \{0,1\} \times E_d^k$.*

Since the input tape cannot be rewritten we need no new symbol for its current tape cell. Due to the same fact δ may only expect symbols from $A \cup \{\sqcup\}$ on the input tape. The set of rejecting states is implicitly given by the partitioning, i.e., $S \setminus F$. The unit vectors correspond to the possible moves of the read-write heads.

Let $\mathcal{M}$ be an NTM_k^d. A *configuration* of $\mathcal{M}$ is a description of its global state which is a $(2(k+1)+1)$-tuple $(s, f_0, f_1, \ldots, f_k, p_0, p_1, \ldots, p_k)$ where $s \in S$ is

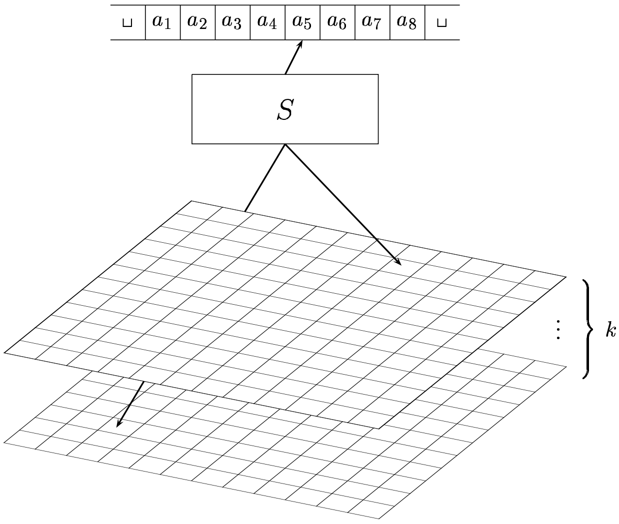

Figure 1: Two-dimensional Turing machine with k working tapes and an input tape.

the current state, $f_0 : \mathbb{Z} \to A \cup \{\sqcup\}$ and $f_i : \mathbb{Z}^d \to T$ are functions that map the tape cells of the corresponding tape to their current contents, and $p_0 \in \mathbb{Z}$ and $p_i \in \mathbb{Z}^d$ are the current head positions, $1 \leq i \leq k$.

The initial configuration $(s_0, f_0, f_1, \dots, f_k, 1, 0, \dots, 0)$ at time 0 is defined by the input word $w = a_1 \cdots a_n \in A^*$, the initial state s_0 and blank working tapes:

$$\begin{aligned} f_0(m) &= \begin{cases} a_m & \text{if } 1 \leq m \leq n \\ \sqcup & \text{otherwise} \end{cases} \\ f_i(m_1, \dots, m_d) &= \sqcup \quad \text{for } 1 \leq i \leq k \end{aligned}$$

Successor configurations are nondeterministically chosen according to the *global transition function* Δ: Let $(s, f_0, f_1, \dots, f_k, p_0, p_1, \dots, p_k)$ be a configuration. Then

$$(s', f_0, f'_1, \dots, f'_k, p'_0, p'_1, \dots, p'_k) \in \Delta(s, f_0, f_1, \dots, f_k, p_0, p_1, \dots, p_k)$$

if and only if there exists

$$(s', x_1, \dots, x_k, j_0, j_1, \dots, j_k) \in \delta(s, f_0(p_0), f_1(p_1), \dots, f_k(p_k))$$

such that

$$\begin{aligned} f'_i(m_1, \dots, m_d) &= \begin{cases} f_i(m_1, \dots, m_d) & \text{if } (m_1, \dots, m_d) \neq p_i \\ x_i & \text{if } (m_1, \dots, m_d) = p_i \end{cases} \\ p'_i &= p_i + j_i, \quad p'_0 = p_0 + j_0 \end{aligned}$$

for $1 \leq i \leq k$. Thus, the global transition function Δ is induced by δ.

Throughout the paper we are dealing with so-called multitape machines:

$$\mathrm{NTM}^d = \bigcup_{k \in \mathbb{N}} \mathrm{NTM}^d_k$$

A single transition step is called *nondeterministic* if the machine has more than one choice for its next move. Otherwise the transition is called *deterministic*. We obtain a natural way to measure the nondeterminism of a computation simply by counting the number of its nondeterministic transitions. In the sequel we refine the nondeterminism by placing bounds on the number of allowed nondeterministic transitions. Clearly, the bounding functions themselves will be bounded by the time complexity. Let $g : \mathbb{N}_0 \to \mathbb{N}_0$ be a function. A d-dimensional Turing machine whose number of nondeterministic transitions for all possible computations on all inputs w is bounded by $g(|w|)$ is denoted by g-NTM^d. For the constant function $g(n) = 0$ we obtain computations that can be performed by *deterministic* Turing machines (DTM^d). In [4] the definition of Turing machines that obey the bound g is in some sense weaker. There the minimum number of nondeterministic transitions made during any computation that accepts w is taken. Since weaker bounds might yield more powerful devices here we remark that all of the following negative results remain valid even in the weak case. This is obvious for positive results.

A Turing machine $\mathcal{M} = \langle S, T, A, \delta, s_0, F\rangle$ *halts* iff the transition function is undefined for the current configuration. An input word $w \in A^*$ is *accepted* by $\mathcal{M}$ if the machine halts at some time in an accepting state, otherwise it is rejected. $L(\mathcal{M}) = \{w \in A^* \mid w \text{ is accepted by } \mathcal{M}\}$ is the *language accepted* by $\mathcal{M}$. Let $t : \mathbb{N}_0 \to \mathbb{N}$, $t(n) \geq n + 1$, be a function, then $\mathcal{M}$ is said to be *t-time-bounded* or of *time complexity t* iff it halts for all possible computations on all inputs w after at most $t(|w|)$ time steps. If t equals the function $id + 1$ acceptance is said to be in *real-time*. The *linear-time* languages are defined according to time complexities $t = c \cdot id$ where $c \in \mathbb{Q}$ with $c \geq 1$. Since time complexities are mappings to positive integers and have to be greater than or equal to $id + 1$, actually, $c \cdot id$ means $\max\{\lceil c \cdot id\rceil, id + 1\}$. Accordingly, for functions that bound the nondeterminism $g(n)$ actually means $\lceil g(n)\rceil$. But for convenience we simplify the notation in the sequel.

The family of all languages which can be accepted by g-NTM^d with time complexity t is denoted by $g\text{-}\mathsf{TIME}^d(t)$. For NTM^d resp. DTM^d language families we use the corresponding notation $\mathsf{NTIME}^d(t)$ resp. $\mathsf{DTIME}^d(t)$.

In order to prove tight hierarchies dependent on bounding functions in general honest functions are required. Usually the notion honest is concretized in terms of computability or constructibility of the functions with respect to the device in question.

Definition 2 *Let $d \in \mathbb{N}$ be a constant. A function $g : \mathbb{N}_0 \to \mathbb{N}_0$ is said to be* DTM^d-constructible *iff there exists a* DTM^d *which for every $n \in \mathbb{N}$ on input 1^n halts after exactly $g(n)$ time steps.*

Another common definition of constructibility demands the existence of an $O(f)$-time-bounded Turing machine that computes the binary representation of the value $g(n)$ on input 1^n. Both definitions have been proven to be equivalent for multitape machines [15].

The following definition summarizes the properties of the honest functions used throughout the paper and names them.

Definition 3 *The set of all functions g with the property $g(m \cdot n) \leq g(m) + g(n)$ whose inverses are increasing, unbounded* DTM^d*-constructible functions is denoted by* $\mathscr{T}^{-1}(\text{DTM}^d)$. *Accordingly, the set of functions* $\{g^{-1} \mid g \in \mathscr{T}^{-1}(\text{DTM}^d)\}$ *is denoted by* $\mathscr{T}(\text{DTM}^d)$.

Since we are interested in a refinement of the nondeterminism we need small bounding functions below the logarithm. The constructible functions are necessarily greater than the identity. Therefore, the inverses of constructible functions are used. The properties increasing and unbounded are straightforward. At first glance the property $g(m \cdot n) \leq g(m) + g(n)$ seems to be restrictive, but it is not. It is easily verified that almost all of the commonly considered bounding functions below the logarithm have this property. As usual here we remark that even the family $\mathscr{T}(\text{DTM}^1)$ is very rich. More details can be found for example in [1, 20].

Due to the small time bounds the devices under investigation are too weak for diagonalization. In order to separate complexity classes counting arguments are used. The following equivalence relation is well-known. At least implicitly it has been used several times in connection with real-time computations, e.g. in [8, 16] for Turing machines and in [3] for iterative arrays.

Definition 4 *Let $L \subseteq A^*$ be a language over an alphabet A and $l \in \mathbb{N}_0$ be a constant. Two words w and w' are l-equivalent with respect to L if and only if*

$$ww_l \in L \iff w'w_l \in L$$

for all $w_l \in A^l$. The number of l-equivalence classes of words of length $n - l$ with respect to L (i.e. $|ww_l| = n$) is denoted by $N(n, l, L)$.

The underlying idea is to bound the number of distinguishable equivalence classes. The following lemma gives a necessary condition for a language to be $(id + r)$-time acceptable by a g-NTM^d.

Lemma 5 *Let $r : \mathbb{N}_0 \to \mathbb{N}$ and $g : \mathbb{N}_0 \to \mathbb{N}_0$ be two functions and $d \in \mathbb{N}$ be a constant. If $L \in g\text{-}\mathsf{TIME}^d(id + r)$, then there exist constants $p, q \in \mathbb{N}$ such that*

$$N(n, l, L) \leq p^{(l+r(n))^d \cdot q^{g(n)}}.$$

Proof. Let $\mathcal{M} = \langle S, T, A, \delta, s_0, F \rangle$ be a $(id + r)$-time g-NTM^d that accepts a language L.

In order to determine an upper bound for the number of l-equivalence classes we consider the possible situations of $\mathcal{M}$ after reading all but l input symbols. The remaining computation depends on the current internal state and the contents of the at most $(2(l + r(n)) + 1)^d$ cells on each tape that are still reachable during the last at most $l + r(n)$ time steps.

Let $p_1 = \max\{|T|, |S|\}$.

For the $(2(l+r(n))+1)^d$ cells per tape there are at most $p_1^{(2(l+r(n))+1)^d}$ different inscriptions. For some $k \in \mathbb{N}$ tapes we obtain altogether at most $p_1^{k(2(l+r(n))+1)^d+1}$ different situations. For $p = p_1^{(k+1)\cdot 3^d}$ these are less than $p^{(l+r(n))^d}$.

Now let q be the maximal number of choices that $\mathcal{M}$ has for any nondeterministic transition step. It follows that the number of different computation paths is at most $q^{g(n)}$. Since the number of equivalence classes is not affected by the last l input symbols in total there are at most $(p^{(l+r(n))^d})^{q^{g(n)}}$ classes. □

The next result is somehow technical but it is a helpful tool for calculating the order of magnitude of the lower bound for the number of induced equivalence classes.

Lemma 6 *Let $f : \mathbb{N}_0 \to \mathbb{N}$ and $g : \mathbb{N}_0 \to \mathbb{N}$ be two functions such that $\exists\, n_0 \in \mathbb{N} : \forall\, n \geq n_0 : \log(g(n)) \leq \frac{1}{4} f(n)$, then*

$$\binom{2^f}{g} \in 2^{\Omega(f\cdot g)}.$$

Proof.

$$\binom{2^f}{g} \geq \left(\frac{2^f - g}{g}\right)^g \geq \left(\frac{2^{\frac{1}{2}f}}{g}\right)^g \geq \left(2^{\frac{1}{2}f - \log(g)}\right)^g \geq \left(2^{\frac{1}{4}f}\right)^g = 2^{\frac{1}{4}f\cdot g} \in 2^{\Omega(f\cdot g)}$$

□

3 Nondeterministic Hierarchies Below Linear-Time

In this section we will present nondeterministic hierarchies for time complexities in the range between real-time and $id + id^{\frac{1}{2}}$. Our result covers the hierarchy for the special case of real-time computations that have been proved in [4]. Since throughout the section we fix the dimension to be 1 we write g-NTM, f-TIME etc. instead of g-NTM^1, f-TIME^1 etc.

Theorem 7 *Let $f : \mathbb{N}_0 \to \mathbb{N}_0$, $g : \mathbb{N}_0 \to \mathbb{N}_0$ and $r : \mathbb{N}_0 \to \mathbb{N}$ be increasing functions. If $f \in \mathscr{T}^{-1}(\text{DTM})$, $f \in O(\log)$, $g \in o(f)$ and $r \in O(id^{\frac{1}{2}})$, then*

$$f\text{-TIME}(id) \setminus g\text{-TIME}(id + r) \neq \emptyset.$$

Proof. At first we define a witness language L_1 for the assertion. The words of L_1 are of the form

$$a^m b^{f^{-1}(m)} c^{2^m} w_1 \$ \cdots \$ w_{2^m} ¢ y$$

where $m \in \mathbb{N}$ is a positive integer, the words $y, w_i \in \{0,1\}^+$ such that $|w_i| = |y| = f^{-1}(m)$ for $1 \leq i \leq 2^m$. In addition, there exists at least one subword w_i that matches the reversal y^R of y.

The basic idea for the construction of an accepting real-time f-NTM $\mathcal{M}$ is as follows. In a first phase $\mathcal{M}$ guesses the position i of the matching subword. Then the corresponding subword is stored on a tape and, finally, is compared to y. Special attention has to be paid such that $\mathcal{M}$ obeys the bound f for any input. More detailed:

The machine $\mathcal{M}$ reads the m symbols a and stores them onto a tape. When the first b appears in the input a constructor for $f^{-1}(m)$ is simulated in order to verify the number of b's. Up to now no nondeterministic transitions have been performed. Moreover, due to the verified number of b's the machine $\mathcal{M}$ may in any case perform m nondeterministic transitions.

While reading the c's $\mathcal{M}$ sequentially simulates constructors for $2^0, 2^1, \dots, 2^{m-1}$. This takes $2^0 + 2^1 + \dots + 2^{m-1} = 2^m - 1$ time steps and is, therefore, well suited to verify the number of c's.

In addition parallel to the first step of a construction a bit is guessed. This bounds the number of nondeterministic transitions appropriately to m. The sequence of guessed bits is a binary number between 0 and $2^m - 1$ that selects the matching subword w_i. $\mathcal{M}$ needs this number in unary on one of its working tapes. Therefore, if the guessed bit is 1, during the subsequent construction 1's are written. If the guessed bit is 0, then nothing is written on that tape.

During the last phase $\mathcal{M}$ simply deletes one 1 for each w_i appearing in the input. When the last 1 has been deleted the subword is stored on a tape and is (later on) compared to the y in reverse order. We conclude that $\mathcal{M}$ works in real-time and obeys the bound f and, hence, $L_1 \in f\text{-}\mathsf{TIME}(id)$.

The second part of the proof is to show $L_1 \notin g\text{-}\mathsf{TIME}(id + r)$. The idea is to approximate the numbers of induced and distinguishable equivalence classes and to show their different growth rates. Therefore, in contrast to the assertion we assume L_1 is acceptable within time $id + r$ by some g-NTM.

For the lengths n of the words in L_1 we obtain:

$$\begin{aligned} n &= m + f^{-1}(m) + 2^m + 2^m \cdot (f^{-1}(m) + 1) + f^{-1}(m) \\ &\leq f^{-1}(m) + 2^m \cdot (f^{-1}(m) + 3) + f^{-1}(m) \\ &\leq (2^m + 2) \cdot (f^{-1}(m) + 3) \end{aligned}$$

Since $f \in O(\log)$ we have $f^{-1} \in \Omega(2^{id})$ and, thus, $n \leq c_1(f^{-1}(m))^2$ for some $c_1 \in \mathbb{N}$.

Due to Lemma 5 for constants $p, q \in \mathbb{N}$ the number of equivalence classes distinguishable by $\mathcal{M}$ is as follows:

$$
\begin{aligned}
&N(n, f^{-1}(m), L_1)\\
&\leq p^{(f^{-1}(m)+r(n))} \cdot q^{g(n)}\\
&\leq p^{(f^{-1}(m)+r(c_1 \cdot f^{-1}(m)^2))} \cdot q^{g(c_1 \cdot f^{-1}(m)^2)} \quad \text{since } r, g \text{ are increasing}\\
&\leq p^{(f^{-1}(m)+r(c_1^2 \cdot f^{-1}(m)^2))} \cdot q^{o(f(c_1 \cdot f^{-1}(m)^2))}\\
&\leq p^{(f^{-1}(m)+c_2 \cdot f^{-1}(m))} \cdot q^{o(f(c_1)+f(f^{-1}(m))+f(f^{-1}(m)))}\\
&\qquad \text{for some } c_2 \in \mathbb{N} \text{ since } r \in O(id^{\frac{1}{2}}) \text{ and } f(x \cdot y) \leq f(x) + f(y)\\
&\leq p^{(c_3 \cdot f^{-1}(m))} \cdot q^{o(c_4 + 2 \cdot m)} \quad \text{for some } c_3, c_4 \in \mathbb{N}\\
&\leq p^{O(f^{-1}(m))} \cdot q^{o(m)}\\
&\leq p^{O(f^{-1}(m)) \cdot o(2^m)}\\
&\leq p^{o(f^{-1}(m) \cdot 2^m)} \leq 2^{o(f^{-1}(m) \cdot 2^m)}
\end{aligned}
$$

On the other hand, consider two different words

$$a^m b^{f^{-1}(m)} c^{2^m} w_1 \$ \cdots \$ w_{2^m} ¢ \quad \text{and} \quad a^m b^{f^{-1}(m)} c^{2^m} w'_1 \$ \cdots \$ w'_{2^m} ¢.$$

They are not $f^{-1}(m)$-equivalent with respect to L_1 if the sets $\{w_1, \ldots, w_{2^m}\}$ and $\{w'_1, \ldots, w'_{2^m}\}$ are not equal. Therefore, L_1 induces at least

$$N(n, f^{-1}(m), L_1) \geq \binom{2^{f^{-1}(m)}}{2^m}$$

equivalence classes. Since $f^{-1} \in \Omega(2^{id})$ we have $\log(2^m) = m \leq \frac{1}{4} f^{-1}(m)$ and may apply Lemma 6 such that

$$N(n, f^{-1}(m), L_1) \geq 2^{\Omega(f^{-1}(m) \cdot 2^m)}.$$

From the contradiction $L_1 \notin g\text{-}\mathsf{TIME}(id + r)$ follows. □

In general, a logarithmic upper bound for the number of nondeterministic transitions seems quite natural, since all positions of subwords or symbols or whatever can be guessed in binary. An upper bound of $id^{\frac{1}{2}}$ for the time beyond real-time appears not very natural. Unfortunately, we have no result for the range between $id + id^{\frac{1}{2}}$ and linear-time that could shed some light on that bound. The previous theorem says also that below time $id + id^{\frac{1}{2}}$ nondeterminism cannot be saved at the cost of time. So, in some sense nondeterminism is a more powerful resource than time. Thus, between two bounds for the nondeterminism there fits a certain amount of time without affecting the language acceptance power.

Example 8 Since $\mathscr{T}(\text{DTM})$ contains 2^{id} and is closed under composition the functions $\log^{[i]}$ for $i \geq 1$ are belonging to $\mathscr{T}^{-1}(\text{DTM})$. They satisfy all conditions of Theorem 7.

$$\begin{array}{ccccccccc}
\mathsf{DTIME}(id+\log) & \subset \cdots \subset & \log^{[3]}\text{-}\mathsf{TIME}(id+\log) & \subset & \log^{[2]}\text{-}\mathsf{TIME}(id+\log) & \subset & \log\text{-}\mathsf{TIME}(id+\log) \\
\cup & & & & & & \\
\mathsf{DTIME}(id+\log^{[2]}) & \subset \cdots \subset & \log^{[3]}\text{-}\mathsf{TIME}(id+\log^{[2]}) & \subset & \log^{[2]}\text{-}\mathsf{TIME}(id+\log^{[2]}) & \subset & \log\text{-}\mathsf{TIME}(id+\log^{[2]}) \\
\cup & & & & & & \\
\mathsf{DTIME}(id+\log^{[3]}) & \subset \cdots \subset & \log^{[3]}\text{-}\mathsf{TIME}(id+\log^{[3]}) & \subset & \log^{[2]}\text{-}\mathsf{TIME}(id+\log^{[3]}) & \subset & \log\text{-}\mathsf{TIME}(id+\log^{[3]}) \\
\cup & & & & & & \\
\vdots & & & & & & \\
\cup & & & & & & \\
\mathsf{DTIME}(id) & \subset \cdots \subset & \log^{[3]}\text{-}\mathsf{TIME}(id) & \subset & \log^{[2]}\text{-}\mathsf{TIME}(id) & \subset & \log\text{-}\mathsf{TIME}(id)
\end{array}$$

The infinite time hierarchy for the deterministic classes has been shown in [14].

4 Nondeterminism versus Dimension

Relating time and nondeterminism in the previous section we have fixed the third resource under consideration to its minimum, i.e., the dimension to 1. Now we are going to relate nondeterminism and dimension and fix the minimal time, i.e., real-time. It will turn out that there exist two-dimensional hierarchies. An example for the iterated logarithm is depicted in Figure 2.

$$\begin{array}{ccccccccc}
\log\text{-}\mathsf{TIME}(id) & & \log\text{-}\mathsf{TIME}^2(id) & & \log\text{-}\mathsf{TIME}^3(id) & & \log\text{-}\mathsf{TIME}^4(id) & & \cdots \\
\cup & & \cup & & \cup & & \cup & & \\
\log^{[2]}\text{-}\mathsf{TIME}(id) & \subset & \log^{[2]}\text{-}\mathsf{TIME}^2(id) & \subset & \log^{[2]}\text{-}\mathsf{TIME}^3(id) & \subset & \log^{[2]}\text{-}\mathsf{TIME}^4(id) & \subset & \cdots \\
\cup & & \cup & & \cup & & \cup & & \\
\log^{[3]}\text{-}\mathsf{TIME}(id) & \subset & \log^{[3]}\text{-}\mathsf{TIME}^2(id) & \subset & \log^{[3]}\text{-}\mathsf{TIME}^3(id) & \subset & \log^{[3]}\text{-}\mathsf{TIME}^4(id) & \subset & \cdots \\
\cup & & \cup & & \cup & & \cup & & \\
\vdots & & \vdots & & \vdots & & \vdots & & \\
\cup & & \cup & & \cup & & \cup & & \\
\mathsf{DTIME}(id) & \subset & \mathsf{DTIME}^2(id) & \subset & \mathsf{DTIME}^3(id) & \subset & \mathsf{DTIME}^4(id) & \subset & \cdots
\end{array}$$

Figure 2: Two-dimensional hierarchy for the iterated logarithm.

4.1 Nondeterministic Hierarchies for Any Dimension

At first we prove the ‘column hierarchies’, i.e., nondeterministic hierarchies for arbitrary dimensions.

Theorem 9 *Let $f : \mathbb{N}_0 \to \mathbb{N}_0$ and $g : \mathbb{N}_0 \to \mathbb{N}_0$ be two increasing functions and $d \in \mathbb{N}$ be a constant. If $f \in \mathscr{T}^{-1}(\mathrm{DTM}^d)$, $f \in O(\log)$ and $g \in o(f)$, then*

$$g\text{-}\mathsf{TIME}^d(id) \subset f\text{-}\mathsf{TIME}^d(id).$$

Proof. Basically, the witness language is of the same structure as in the previous proof. But since dimensions are a powerful resource a refined control of the length and the number of the subwords is necessary. In addition, an accepting machine needs some hints in order to obey the given bounds. The length of the subwords must not exceed a maximum with respect to a bounding value. Define

$$u_d(m) = \max\{v \in \mathbb{N} \mid (v^{d-1} \cdot 2^m)(v+1) \cdot 2 + v \cdot d \leq m^{d+1} \cdot f^{-1}(m)\}.$$

Observe that $u_d(m)$ is increasing and unbounded since $f^{-1}(m)$ is increasing and unbounded. Now the words of the language L_2 are of the form

$$a^m b^{f^{-1}(m)} c^{2^m} w_1 \$ w_1^R ¢ w_2 \$ w_2^R ¢ \cdots ¢ w_l \$ w_l^R ¢ d_1 \cdots d_j y$$

where $m \in \mathbb{N}$ is a positive integer, $y, w_i \in \{0,1\}^+$ such that $|y| = |w_i| \leq u_d(m)$ for $1 \leq i \leq l$ and $l = |w_1|^{d-1} \cdot 2^m$. With $j = (d-1)|w_1|$ we require $d_1, \dots, d_j \in E_{d-1}$.

The acceptance of such a word is best described by the behavior of an accepting f-NTM^d $\mathcal{M}$.

The behavior of $\mathcal{M}$ until the first subword w_1 appears in the input has been described in the proof of Theorem 9. Parallel to what follows $\mathcal{M}$ verifies the lengths of the w_i and y to be equal. Since the polynomials are constructible and the constructible functions are closed under multiplication $\mathcal{M}$ can check the number l to be $u^{d-1} \cdot 2^m$ and the number j to be $(d-1) \cdot u$. The value $f^{-1}(m)$ is given by the number of b's. Therefore, $\mathcal{M}$ can simulate a constructor for $m^{d+1} \cdot f^{-1}(m)$ and, hence, check that $|w_1 \$ w_1^R ¢ \cdots ¢ w_l \$ w_l^R ¢ d_1 \cdots d_j y| \leq m^{d+1} \cdot f^{-1}(m)$ what implies $u \leq u_d(m)$.

During its first phase $\mathcal{M}$ has guessed a value in the range between 0 and $2^m - 1$ with m nondeterministic transitions. This value selects a block of u^{d-1} consecutive subwords $w_i \$ w_i^R ¢$. Clearly, the guessed value can be decreased correspondingly in order to detect the time step at which the guessed block appears in the input. Now $\mathcal{M}$ stores the subwords w_i of the block in a d-dimensional area of size u^d. If, for example, the head of the corresponding tape is located at the coordinates $(j_1, \dots, j_{d-1}, 0)$ then the following subword w_i is stored into the cells $(j_1, \dots, j_{d-1}, 0), (j_1, \dots, j_{d-1}, 1), \dots, (j_1, \dots, j_{d-1}, u-1)$. Subsequently $\mathcal{M}$ moves the head back to position $(j_1, \dots, j_{d-1}, 0)$ while reading and verifying w_i^R. While reading the following symbol ¢ the head changes to the new coordinates.

The last phase leads to acceptance or rejection. After storing all subwords of the block $\mathcal{M}$ keeps the head on the current position until d_1 appears in the input. While reading the d_i $\mathcal{M}$ moves the head by adding d_i to the current position. Since $d_i \in E_{d-1}$ the dth coordinate is not affected. This phase leads to a head position $(j_1, \dots, j_{d-1}, 0)$. Now the subword y is read and compared to the subword w_i stored at that position (if there is stored a subword at all). $\mathcal{M}$ accepts if and only if both are equal. Altogether $\mathcal{M}$ works in real-time and obeys the bound f for the nondeterministic transitions. It follows $L_2 \in f\text{-}\mathsf{TIME}^d(id)$.

Now assume L_2 is real-time acceptable by some g-NTM^d $\mathcal{M}'$. A contradiction for the numbers of induced and distinguishable equivalence classes is derived as follows.

Two words

$$a^m b^{f^{-1}(m)} c^{2^m} w_1 \$ w_1^R ¢ \cdots ¢ w_l \$ w_l^R ¢$$

and

$$a^m b^{f^{-1}(m)} c^{2^m} w_1' \$ w_1'^R ¢ \cdots ¢ w_l' \$ w_l'^R ¢$$

are not $(d \cdot u)$-equivalent with respect to L_2 if the sets $\{w_1, \ldots w_l\}$ and $\{w_1', \ldots, w_l'\}$ are not equal. There are $\binom{2^u}{u^{d-1} \cdot 2^m}$ different subsets of $\{0,1\}^u$ with $u^{d-1} \cdot 2^m$ elements. Consider words with $u = u_d(m)$. We obtain at least

$$N(n, u \cdot d, L_2) \geq \binom{2^u}{u^{d-1} \cdot 2^m}$$

equivalence classes. From the definition of $u_d(m)$ follows $u^d \cdot 2^m \in \Omega(m^{d+1} \cdot f^{-1}(m))$.

Since f is of order $O(\log)$ its inverse f^{-1} belongs to $\Omega(2^{id})$. This implies $u^d \cdot 2^m \in \Omega(m^{d+1} \cdot 2^m)$ and, thus, $u^d \in \Omega(m^{d+1})$ and, hence, $m \in o(u)$. We conclude $(d-1) \cdot \log(n) + m \leq \frac{1}{4} u$ and so $\log(u^{d-1} \cdot 2^m) \leq \frac{1}{4} u$. Now we can apply Lemma 6 and obtain

$$N(n, u \cdot d, L_2) \geq 2^{\Omega(u^d \cdot 2^m)}.$$

In order to bound the number of distinguishable equivalence classes we approximate g as follows:

$$\begin{aligned} g(n) &\leq (m + f^{-1}(m) + 2^m + m^{d+1} \cdot f^{-1}(m)) \quad \text{since } g \text{ is increasing} \\ &\leq g(f^{-1}(m) \cdot (m^{d+1} + 3)) \quad \text{since } f^{-1}(m) \in \Omega(2^m) \\ &\leq g((f^{-1}(m))^2) \\ &\leq o(f((f^{-1}(m))^2)) \\ &\leq o(f(f^{-1}(m)) + f(f^{-1}(m))) \quad \text{since } f(x \cdot y) \leq f(x) + f(y) \\ &\leq o(2 \cdot m) = o(m) \end{aligned}$$

Finally, we conclude

$$\begin{aligned} N(n, u \cdot d, L_2) &\leq p^{(u \cdot d)^d \cdot q^{g(n)}} \leq p^{O(u)^d \cdot q^{o(m)}} \\ &\leq p^{O(u^d) \cdot o(2^m)} \leq 2^{o(u^d \cdot 2^m)} \end{aligned}$$

From the contradiction $L_2 \notin g\text{-}\mathsf{TIME}^d(id)$ follows what completes the proof. □

4.2 Dimension Hierarchies for Refined Nondeterminism

Our last result concerns the 'line hierarchies' of the previous example, i.e., dimension hierarchies for the bounds of nondeterminism in question. The proof follows the ideas of the previous results but the witness language is simpler.

It will turn out that dimensions are a very powerful resource since the witness language is even acceptable by a deterministic Turing machine if an additional dimension is provided.

Theorem 10 *Let $g : \mathbb{N}_0 \to \mathbb{N}_0$ be an increasing function and $d \in \mathbb{N}$ be a constant. If $g \in o(\log)$ then*

$$\mathsf{DTIME}^{d+1}(id) \setminus g\text{-}\mathsf{TIME}^{d}(id) \neq \emptyset.$$

Proof. The words of a language L_3 are of the form

$$w_1 \$ w_1^R ¢ \cdots ¢ w_l \$ w_l^R ¢ d_1 \cdots d_j y$$

where $u \in \mathbb{N}$ is a positive integer such that $y, w_i \in \{0,1\}^u$ for $1 \leq i \leq l$, and $l = u^d$. With $j = d \cdot u$ we require $d_1, \dots, d_j \in E_d$.

A $(d+1)$-dimensional deterministic real-time acceptor $\mathcal{M}$ for L_3 works similarly as described in the previous proof. Here the length $d \cdot u$ and the number u^d can be verified by constructors that operate with the length of w_1 as argument. Moreover, all subwords w_i are stored in a $(d+1)$-dimensional area of size u^{d+1}. Again, an input is accepted if the y matches the subword which is stored at the position that is reached by interpreting the d_i.

The lower bound for the number of induced equivalence classes is once more due to the observation that two words are not equivalent if their sets of subwords w_i are different.

Since $\log(u^d) = d \cdot \log(u) \leq \frac{1}{4}u$ by applying Lemma 6 we obtain

$$N(n, u \cdot d + u, L_3) \geq \binom{2^u}{u^d} \geq 2^{\Omega(u^{d+1})}.$$

For the number of real-time g-NTM^d distinguishable equivalence classes we calculate for $p, q \in \mathbb{N}$:

$$\begin{aligned}
N(n, u \cdot d + u, L_3) &\leq p^{(d \cdot u + u)^d \cdot q^{g((2u+2) \cdot u^d + u + d \cdot u)}} \\
&\leq p^{O(u)^d \cdot q^{g(c_1 \cdot u^{d+1})}} && \text{for some } c_1 \in \mathbb{N} \\
&\leq p^{O(u^d) \cdot q^{o(\log(c_1 \cdot u^{d+1}))}} && \text{since } g \text{ is increasing} \\
&\leq p^{O(u^d) \cdot q^{o(O(\log(u)))}} \\
&\leq p^{O(u^d) \cdot q^{o(\log(u))}} \\
&\leq p^{O(u^d) \cdot o(u)} \\
&\leq p^{o(u^{d+1})} = 2^{o(u^{d+1})}
\end{aligned}$$

Therefore, L_3 cannot belong to g-$\mathsf{TIME}^d(id)$ and the assertion follows. □

References

[1] J. L. BALCÁZAR, J. DÍAZ, J. GABARRÓ, *Structural Complexity I*. Springer, Berlin, 1988.

[2] L. CAI, J. AND CHEN, On the amount of nondeterminism and the power of verifying. *SIAM Journal on Computing* **26** (1997), 733–750.

[3] S. N. COLE, Real-time computation by n-dimensional iterative arrays of finite-state machines. *IEEE Transactions on Computers* **C-18** (1969), 349–365.

[4] P. C. FISCHER, C. M. R. KINTALA, Real-time computations with restricted nondeterminism. *Mathematical Systems Theory* **12** (1979), 219–231.

[5] J. GOLDSMITH, M. A. LEVY, M. MUNDHENK, Limited nondeterminism. *Sigact News* **27** (1996), 20–29.

[6] J. GOLDSTINE, C. M. R. KINTALA, D. WOTSCHKE, On measuring nondeterminism in regular languages. *Information and Computation* **86** (1990), 179–194.

[7] J. GOLDSTINE, H. LEUNG, D. WOTSCHKE, On the relation between ambiguity and nondeterminism in finite automata. *Information and Computation* **100** (1992), 261–270.

[8] J. HARTMANIS, R. E. STEARNS, On the computational complexity of algorithms. *Transactions of the AMS* **117** (1965), 285–306.

[9] C. HERZOG, Pushdown automata with bounded nondeterminism and bounded ambiguity. *Theoretical Computer Science* **181** (1997), 141–157.

[10] J. HROMKOVIC ET AL., Measures of nondeterminism in finite automata. In: *Proc. Int. Conf. on Automata, Languages, and Programming (ICALP)*. LNCS **1853**, Springer, 2000, 199–210.

[11] C. M. R. KINTALA, Computations with a restricted number of nondeterministic steps. Ph.D. thesis, Pennsylvania State University, 1977.

[12] C. M. R. KINTALA, P. C. FISCHER, Refining nondeterminism in relativized complexity classes. *SIAM Journal on Computing* **13** (1984), 329–337.

[13] C. M. R. KINTALA, D. WOTSCHKE, Amounts of nondeterminism in finite automata. *Acta Informatica* **13** (1980), 199–204.

[14] A. KLEIN, M. KUTRIB, Deterministic Turing machines in the range between real-time and linear-time. *Theoretical Computer Science* (2002), to appear.

[15] K. KOBAYASHI, On proving time constructibility of functions. *Theoretical Computer Science* **35** (1985), 215–225.

[16] A. L Rosenberg, Real-time definable languages. *Journal of the ACM* **14** (1967), 645–662.

[17] K. Salomaa, S. Yu, Limited nondeterminism for pushdown automata. *Bulletin of the EATCS* **50** (1993), 186–193.

[18] K. Salomaa, S. Yu, Measures of nondeterminism for pushdown automata. *Journal of Computer and System Sciences* **49** (1994), 362–374.

[19] D. Vermeir, W. Savitch, In the amount of nondeterminism in pushdown automata. *Fundamenta Informaticae* **4** (1981), 401–418.

[20] K. Wagner, G. Wechsung, *Computational Complexity.* Reidel, Dordrecht, 1986.