MÉMOIRES DE LA S. M. F.

HUGO VOLGER

The role of rudimentary relations in complexity theory

Mémoires de la S. M. F. 2e série, tome 16 (1984), p. 41-51

<http://www.numdam.org/item?id=MSMF_1984_2_16__41_0>

Société Mathématique de France
2° série, mémoire n° 16, 1984, p. 41-51

THE ROLE OF RUDIMENTARY RELATIONS IN COMPLEXITY THEORY

Volger, Hugo

Résumé:

On étudie dans cet article les classes R et XR des relations rudimentaires et faiblement rudimentaires qui se reposent sur la relation de la concaténation bornée. On obtient RUD et XRUD, les classes correspondantes des langages, comme l'union d'une hiérarchie linéaire resp. polynômiale. Ces hiérarchies utilisent des quanteurs alternants aux longueurs bornés ou également des machines alternantes de Turing avec alternance constante. Nous allons introduire une autre description utilisant des quanteurs alternants pour des oracles. En plus on obtiendra une chaîne nouvelle des hiérarchies pour tous les niveaux exponentiels, dont l'union sera ERUD, l'analogue exponentiel de la classe RUD. Et on va montrer que ERUD est la classe E_3 des langages élémentaires.

Abstract:

We shall study the classes R resp. XR of rudimentary resp. extended rudimentary relations which are based on the relation of bounded concatenation. The associated classes RUD resp. XRUD of languages are the union of a linear - resp. polynomial time hierarchy. It can be described either by means of alternating length bounded quantifiers or by means of Turing machines with constant alternation. We shall introduce another description based on alternating quantifiers for oracle sets. Extending these results we obtain a chain of hierarchies for the iterated exponential time levels, whose union is the class ERUD, the exponential analogue of RUD. Moreover, it will be shown that ERUD coincides with the class of elementary recursive languages.

Table of contents:

0037-9484/84 03 41 11/\$ 3.10/ © Gauthier-Villars

Hugo Volger

1. Introduction:

This paper is a survey on the classes R , XR , ER of rudimentary resp. extended rudimentary resp. exponential rudimentary relations and the corresponding classes RUD , XRUD , ERUD of languages . R and XR were introduced by Smullyan in 1961 resp. Bennett in 1962 (cf.[19],[1]) , whereas ER is a new class . As we shall see later , a relation is rudimentary if it is definable from the concatenation relation by means of a first order formula where all quantifiers have linear length bounds . XR resp. ER will be the polynomially resp. exponentially bounded analogue of R .

The associated classes RUD , XRUD , ERUD may be obtained as the union of certain hierarchies . In her thesis in 1975 Wrathall [27] has shown that there are length bounded quantification hierarchies which yield LH = RUD resp. PH = XRUD and have as first step NLTIME resp. NPTIME . As length bounded quantification is closely related to time bounded alternation , these hierarchies can also be described as constant alternation hierarchies for LH and PH (cf.Chandra,Stockmeyer [4],Kozen [10]) .

Recently Orponen [16] has introduced a class EH as the union of an exponential time hierarchy involving oracle set quantification and having NEXPTIME as a first step . Extending his approach we are able to describe the hierarchies for LH and PH as oracle set quantification hierarchies . Moreover , we shall introduce classes $EH^{(i)}$ as the union of an analogous hierarchy involving the i-th iterate e_i of the exponential function , and we shall show that each of the three descriptions may be used . As a consequence we obtain that ERUD is the union of the classes $EH^{(i)}$ and coincides with the class of elementary recursive languages . In addition , the alternating log-space hierarchy of Chandra,Kozen and Stockmeyer [5] may be viewed as step -1 of this chain of hierarchies .

The class $EH^{(i)}$ which consists of languages requiring a constant number of alternations is contained in the class LA_i the corresponding class with a linear amount of alternation . Recently we have shown that the decision problem of the theory e_i-bounded concatenation is complete in the class LA_i w.r.t. polynomial time reductions for $i \geq 1$. In a certain sense these results for $EH^{(i)}$ and LA_i measure the power of e_i-bounded concatenation (cf.also Wilkie [24,25,26]) . However , the question whether the inclusion $EH^{(i)} \subseteq LA_i$ is proper for some $i \geq 0$ remains open . A positive answer would imply that the inclusions PH $\subseteq$ APTIME and LH $\subseteq$ ALTIME are proper , thus solving important open problems in complexity theory .

2. Concatenation as a base of computability theory:

In 1946 Quine [17] suggested to use the concatenation relation rather than addition and multiplication as a base of computability theory . Thus in 1961 Smullyan [19] introduced the class R resp. R_s of rudimentary resp. strictly rudimentary relations on {1,2}* . They consist of those relations which are definable from the concatenation relation by a first order formula where all quantifiers have a linear bound resp. are subword quantifiers. Smullyan has shown that R_s is all we need to describe computations. Each language $L \subseteq \{1,2\}^*$ which is recursively enumerable i.e. accepted by some Turing machine M can be obtained from a relation Q in R_s as follows: $x \in L$ iff $\exists y: (x,y) \in Q$, where $(x,y) \in Q$ expresses the fact that y is an accepting computation sequence with input x. This shows that R_s is large enough to enable us to describe Turing machine computations by means of words consisting of sequences of configuration words. On the other hand R_s is quite small since the associated class RUD_s of languages is contained in LOGSPACE and does not contain $\{1^n 2^n : n \in N\}$ (cf. Nepomnjascii [15], Meloul [11]). In addition, the NPTIME-complete problem SAT(x) is of the form $\exists y: |y| \leq |x| \wedge Q(x,y)$ with Q in R_s as Meloul [11] has shown. This may explain why the class R_s and the related classes R and XR play an important role in complexity theory.

3. The rudimentary relations:

The class R resp. R_s of rudimentary resp. strictly rudimentary relations on $\{1,2\}^*$, introduced by Smullyan [19], is defined as the least class of relations which contains the concatenation relation Con and which is closed under the boolean operations, explicit transformations and linearly bounded resp. subword quantification. The class R^+ of positive rudimentary relations on $\{1,2\}^*$, introduced by Bennett [1], is defined as the least class of relations which contains the relation Con and which is closed under finite unions and intersections, explicit transformations, subword quantification and linearly bounded existential quantification.

$\exists y: y \subseteq x \wedge \ldots$, $\forall y: y \subseteq x \rightarrow \ldots$	subword quantification
$\exists y: \lvert y \rvert \leq k \lvert x \rvert \wedge \ldots$, $\forall y: \lvert y \rvert \leq k \lvert x \rvert \rightarrow \ldots$	linearly bounded quantification

Using the k-adic encoding words over $\{1,\ldots,k\}$ may be identified with natural numbers. Bennett [1] has shown that modulo the dyadic encoding R coincides with the class CA of constructive arithmetic relations on N, which is the analogue of R on N using + and × rather than Con. In addition, CA coincides with the class of bounded arithmetic relations of Harrow [6]. Moreover, the analogues of R resp. R_s resp. R^+ on $\{1,\ldots,k\}^*$ coincide with R resp. R_s resp. R^+ on $\{1,2\}^*$ modulo the k-adic encoding and the dyadic decoding. Using the sequential encoding $\theta(Q)$ of a relation Q one obtains the corresponding classes of languages on $\{1,2,\S\}$: RUD, RUD_s, RUD^+. It can be shown that these classes may be identified with the unary relations in R, R_s, R^+.

Replacing linearly bounded quantification by polynomially bounded quantification (i.e. $\exists y: |y| \leq |x|^k \wedge \ldots$ and $\forall y: |y| \leq |x|^k \rightarrow \ldots$) one obtains the classes of extended rudimentary resp. extended positive rudimentary relations, which were introduced by Bennett [1].

Going a step further we introduce the classes ER resp. ER^+ of exponential rudimentary resp. exponential positive rudimentary relations. They are obtained from

R resp. R^+ by replacing linearly bounded quantification by exponentially bounded quantification (i.e. $\exists y: |y| \le e_1(|x|^k) \wedge \ldots$ and $\forall y: |y| \le e_1(|x|^k) \rightarrow \ldots$ with $e_1(n) = 2^n$). Clearly, iterated exponential functions can be used as length bounds as well. - The corresponding classes of languages are denoted by XRUD, $XRUD^+$ resp. ERUD, $ERUD^+$. These classes are related as follows: $RUD_s \subseteq RUD^+ \subseteq RUD$, $XRUD^+ \subseteq XRUD$, $ERUD^+ \subseteq ERUD$ and $RUD^+ \subseteq XRUD^+ \subseteq ERUD^+$, $RUD \subseteq XRUD \subseteq ERUD$.

It should be mentioned that Jones [8] has introduced sublinear analogues of the class R resp. RUD. In particular, he considered a subclass RUD_{log} of LOGSPACE. It is not clear how this class fits into the above set up.

4. Turing machines with constant resp. linear alternation:

Chandra and Stockmeyer [4] and Kozen [10] have extended the concept of nondeterministic Turing machines (NTM's) to alternating Turing machines (ATM's). There is a close connection between alternation and quantification. In particular, hierarchies defined by bounded quantification are closely related to hierarchies defined by constant alternation using the same time bound.

An ATM $\underline{M}$ is a NTM which has 2 disjoint sets of states, the existential and universal states, and a distinguished accepting resp. rejecting state. Configurations and their successor relation are defined as for NTM's. An input w is accepted by $\underline{M}$ (i.e. $w \in L(\underline{M})$), if there exists a finite accepting subtree B of the computation tree of $\underline{M}$ for w. B is accepting, if (1) the root of B is labeled with the input configuration for w, (2) all leaves of B are labeled with accepting configurations, (3) if a node b of B is labeled with an existential (resp. universal) configuration C then at least one (resp. all) successor configurations C' of C must appear as labels of successors b' of b (cf. Berman [2]).

A language L belongs to the alternation class STA(s,t,a), if L is accepted by an ATM $\underline{M}$ such that each w in L posesses an accepting subtree B of depth $\le t(n)$ and alternation depth $\le a(n)$ and each configuration in B uses space $\le s(n)$, where $n=|w|$. We shall use the notation $STA_\exists(s,t,a)$ resp. $STA_\forall(s,t,a)$ to indicate that the input configuration is required to be existential resp. universal. As special cases we obtain the alternating time class $ATIME(t) = STA(-,t,-)$ and the alternating space class $ASPACE(s) = STA(s,-,-)$. The time class with constant alternation CATIME(t) is defined as $\bigcup \langle STA_\exists(-,t,k) : k \in N \rangle$. Similarly the time class with linear alternation LATIME(t) is defined as $STA_\exists(-,t,id)$.

Alternating time bridges the gap between nondeterministic time and deterministic space as Chandra, Kozen and Stockmeyer [5] have shown:

(*) $NTIME(t) \subseteq CATIME(t) \subseteq LATIME(t) \subseteq ATIME(t) \subseteq DSPACE(t)$ for $t \ge id$

(**) ALOGSPACE = PTIME, APTIME = PSPACE, APSPACE = EXPTIME

5. The linear - and polynomial time hierarchies:

Wrathall [27] has shown that the class XRUD is the union of the polynomial time hierarchy of Meyer and Stockmeyer [12], and that the class RUD is the union of a linear time analogue of this hierarchy. There are several descriptions of these two hierarchies as we shall see below.

Constant Alternation:

$APH = U\langle AP_k : k \in N\rangle$, $AP_k = U\langle STA_\exists(-,O(n^i),k) : i \in N\rangle$ for $k \geq 1$, $AP_o = PTIME$,
$ALH = U\langle AL_k : k \in N\rangle$, $AL_k = STA_\exists(-,O(n),k)$ for $k \geq 1$, $AL_o = LTIME$.
Hence we have $APH = U\langle CATIME(O(n^i)) : i \in N\rangle$, $ALH = CATIME(O(n))$

Length Bounded Quantification:

$PH = U\langle P\Sigma_k : k \in N\rangle$, $P\Sigma_o = PTIME$,
$L \in P\Sigma_k$ iff there exists $L' \in P\Sigma_o$ and $m_1, \dots, m_k$ such that:
$$x \in L \text{ iff } \exists y_1 : |y_1| \leq |x|^{m_1} \dots Q_k y_k : |y_k| \leq |x|^{m_k} : (x,y_1,\dots,y_k) \in L' .$$
$LH = U\langle L\Sigma_k : k \in N\rangle$, $L\Sigma_o = LTIME$,
$L \in L\Sigma_k$ iff there exist $L' \in L\Sigma_o$ and $m_1, \dots, m_k$ such that:
$$x \in L \text{ iff } \exists y_1 : |y_1| \leq m_1|x| \dots Q_k y_k : |y_k| \leq m_k|x| : (x,y_1,\dots,y_k) \in L' .$$

Oracle Set Quantification:

$OPH = U\langle OP_k : k \in N\rangle$, $OP_o = U\langle STA_\exists(\log(n^i),-,k) : i,k \in N\rangle$,
$L \in OP_k$ iff there exists a constant alternation oracle TM $\underline{M}$ with k oracles working in space $\log(n^i)$ for some i such that:
$$x \in L \text{ iff } \exists A_1 \dots Q_k A_k : \underline{M} \text{ accepts } x \text{ with the oracles } A_1, \dots, A_k .$$

Iterated Nondeterministic Oracles:

$NP_* = U\langle NP_k : k \in N\rangle$, $NP_o = PTIME$, $NP_{k+1} = \underline{NP}(NP_k)$,
$NL_* = U\langle NL_k : k \in N\rangle$, $NL_o = LTIME$, $NL_{k+1} = \underline{NL}(NL_k)$,
where $\underline{NP}(\underline{A})$ resp. $\underline{NL}(\underline{A})$ is the class of languages accepted by a nondeterministic oracle TM with a polynomial resp. linear time bound and an oracle for a member of $\underline{A}$.

The following 2 propositions show that the union of these hierarchies is XRUD resp. RUD and that all descriptions yield the same hierarchies.

Prop. 1: (1) $NP_k = P\Sigma_k$ for k in N; $NP_* = PH$
(2) $PH = XRUD$; $NP_1 = NPTIME = XRUD^+$
(3) $NL_k = L\Sigma_k$ for k in N; $NL_* = LH$
(4) $LH = RUD$; $NL_1 = NLTIME \subseteq RUD^+$

The proofs of (1) - (4) except $NL_1 \subseteq RUD^+$ can be found in Wrathall [28,29]. An application of a result of Book and Greibach [3] to the inclusion $CFL \subseteq RUD^+$ in Yu [30] yields the desired inclusion (cf. Meloul [11]).

The proof of the next proposition will be given in some detail since the result will be generalized later on.

Prop.2: (1) $AP_k = P\Sigma_k$ for k in N ; APH = PH

(2) $AL_k = L\Sigma_k$ for k in N ; ALH = LH

(3) $OP_k = AP_k$ for $k \geq 1$ in N ; $OP_0 \subseteq AP_0$; OPH = APH .

The result in (1) was mentioned in Chandra , Kozen and Stockmeyer [5] and the analogous result in (2) can be found in Volger [23] . (3) is a new result which constitutes an analogue of a result of Orponen [16] for EH , the union of an exponential time hierarchy .

(1) and (2) can be proved by the same method . Given the syntactic description of L which uses at most k alternations of length bounded quantifiers , it is easy to construct an ATM accepting L with the corresponding time bound and at most k alternations . This proves $P\Sigma_k \subseteq AP_k$ resp. $L\Sigma_k \subseteq AL_k$. - Conversely , given an ATM accepting L with at most k alternations , one constructs a deterministic TM accepting a language L' and having k additional tapes with the following property . Simulating the i-th alternation phase the machine controls the choice of moves to be simulated by reading the i-th tape as long as necessary going from left to right . Hence L can be obtained from L' by an appropriate length bounded quantification with at most k alternations , as desired . This should be compared with the incremental stack automata in Yu [30] . This proves $AP_k \subseteq P\Sigma_k$ resp. $AL_k \subseteq L\Sigma_k$.

To prove (3) we adapt Orponen's proof in [16] . The oracle free part of the constant alternation oracle TM $\underline{M}$ for L can be simulated by a DTM $\underline{M}'$ working in polynomial time because of $STA_{\exists}(\log(n^i),-,k) \subseteq ASPACE(\log(n^i)) \subseteq DTIME(O(n^j))$ for some j . This inclusion can be found in Chandra , Kozen and Stockmeyer [5] . The k quantifiers concerning the oracle sets $A_1 , \dots , A_k$ will be replaced by k alternations of an ATM $\underline{M}''$ extending $\underline{M}'$, where each branch in the j-th alternation phase corresponds to an oracle set $A_j' = A_j \wedge \{1,2\}^{\leq \log(n^1)}$. Because of the space bound of $\underline{M}$ it suffices to consider A_j' instead of A_j . Moreover , each set A_j' can be specified in n^1 steps . Thus $\underline{M}''$ works in polynomial time . This shows $OP_k \subseteq AP_k$.

Conversely , let L be accepted by a constant alternation TM $\underline{M}$ working in polynomial time . The idea is to code a computation sequence α of configurations of $\underline{M}$ by an oracle set $C(\alpha)$ which is coded characterwise . A sequence α of $d = n^1$ configurations of length n^1 is a word of length $\leq d^2$. It can be coded as follows : $C(\alpha) = \{(i,j,\alpha_{i,j}) : i,j \leq d^2\}$, where $\alpha_{i,j}$ is the j-th character in the i-th configuration of α . The indices i , j are short because of $|i|$, $|j| \leq 2\log(n^1)$. Given (i,j) $\alpha_{i,j}$ can be recovered from $C(\alpha)$ by at most a fixed number of queries . Since the successor relation is local , it is possible to construct a constant alternation oracle TM $\underline{M}'$ working on space $\log(n^i)$ for some i such that (u,v) is accepted by $\underline{M}'$ with oracle C iff C codes a computation sequence of $\underline{M}$ starting with u and ending with v and having no alternation except at the last step . Similarly , the input configurations and the accepting configurations can be handled by appropriate machines. In order to express acceptance by the given ATM $\underline{M}$ note that each alternation phase i gives rise to a quantification over an oracle C_i corresponding to it. By this method one obtains a constant alternation oracle TM $\underline{\hat{M}}$ working on space $\log(n^1)$, which does the required job. It should be noted that $\underline{\hat{M}}$ can be chosen to be universal. This shows $AP_k \subseteq OP_k$.

The inclusion $OP_o = \cup\langle STA_\exists(\log(n^1),-,k) : i,k \in N\rangle \subseteq AP_o = PTIME$ follows from PTIME = ALOGSPACE which was proven in Chandra, Kozen and Stockmeyer [5].

6. A chain of exponential time hierarchies:

As mentioned above, Orponen [16] introduced a class EH as the union of an exponential time analogue of the hierarchy for APH = PH. More generally, we shall consider iterated exponential time analogues of the hierarchy for PH and obtain a chain of classes $EH^{(i)}$ whose union is the class $\widetilde{E}$ of elementary recursive languages.

Let e_i be the i-th iterate of the exponential function, i.e. $e_o(n) = n$ and $e_{i+1}(n) = \exp(2,e_i(n))$, where $\exp(2,m) = 2^m$. As before there are several ways of describing the hierarchies for $EH^{(i)}$.

The <u>constant alternation hierarchy</u> $AEH^{(i)} = \cup\langle AE_k^{(i)} : k \in N\rangle$ is obtained from APH by replacing everywhere $O(n^1)$ by $e_i(O(n^1))$. The <u>length bounded quantification hierarchy</u> $EH^{(i)} = \cup\langle E\Sigma_k^{(i)} : k \in N\rangle$ is obtained from PH by replacing everywhere $O(n^1)$ by $e_i(O(n^1))$. The <u>oracle set quantification hierarchy</u> $OEH^{(i)} = \cup\langle OE_k^{(i)} : k \in N\rangle$ is obtained from OPH by replacing everywhere the space bound $\log(n^1)$ by the time bound $e_{i-1}(O(n^1))$ and defining $OE_o^{(i)} = AE$

Orponen [16] considered the hierarchies for $AEH^{(1)}$ and $OEH^{(1)}$ and proved $AEH^{(1)} = OEH^{(1)}$. The hierarchy for $EH^{(1)}$ and all the other hierarchies for $i \geq 2$ seem to be new. In the case $i = 0$ we obtain the hierarchies for APH, PH and OPH discussed earlier. The following proposition extends the results in proposition 2.

<u>Prop.3</u>: For $i \geq 1$ we have:

(1) $AE_k^{(i)} = E\Sigma_k^{(i)}$ for k in N; $AEH^{(i)} = EH^{(i)}$

(2) $OE_k^{(i)} = AE_k^{(i)}$ for $k \geq 1$ in N; $OE_o^{(i)} \subseteq AE_o^{(i)}$; $OEH^{(i)} = AEH^{(i)}$.

This can be proved by the same method which was used to prove (1) and (3) in proposition 2. To prove $OE_o^{(i)} = AEH^{(i-1)} \subseteq AE_o^{(i)}$ we use $STA_\exists(-,e_{i-1}(O(n^1)),k) \subseteq ASPACE(e_{i-1}(O(n^1)) \subseteq DTIME(e_i(O(n^{1'})))$ proved in [5]. Moreover, an oracle set of words of length $e_{i-1}(O(n^1))$ can be specified in $e_i(O(n^{1'}))$ steps, whereas the code of a computation sequence of $e_i(O(n^1))$ configurations of length $e_i(O(n^1))$ uses words of length $\leq e_{i-1}(O(n^{1'}))$. This shows that (1) and (2) can be proved as before.

The next proposition shows that $\widetilde{E}$, the class of <u>elementary recursive languages</u>, coincides with ERUD and that the classes $EH^{(i)}$ form a new hierarchy for $\widetilde{E}$.

We shall use the following abbreviations : $LA_i = U\langle LATIME(e_i(O(n^l))) : l \in N\rangle$ and $AS_i = U\langle ASPACE(e_i(O(n^l))) : l \in N\rangle$.

Prop.4: (1) $AEH^{(i)} \subseteq LA_i \subseteq AS_i \subseteq AE_O^{(i+1)} \subseteq AEH^{(i+1)}$ for i in N

(2) $U\langle AE_O^{(i)} : i \in N\rangle = U\langle AEH^{(i)} : i \in N\rangle = U\langle LA_i : i \in N\rangle = U\langle AS_i : i \in N\rangle = \widetilde{E}$

(3) For each $L \in AEH^{(i)}$ there exists $L' \in LH$ and l in N such that $x \in L$ iff $\exists y : |y| \leq e_i(O(n^l)) : (x,y) \in L'$

(4) $\widetilde{E} = ERUD = ERUD^+$

(5) $AEH^{(i)} \neq AEH^{(i+2)}$; $AE_O^{(i)} \neq AEH^{(i)}$ implies $AEH^{(i)} \neq AEH^{(i+1)}$.

The inclusions needed for (1) can again be found in [5] . (2) is a consequence of (1) because of the well known fact $\widetilde{E} = U\langle AE_O^{(i)} : i \in N\rangle$. To prove the representation result in (3) which represents elements of $EH^{(i)}$ with the help of elements of LH we show (cf.Wrathall [27] in the case $i = 0$) :

(*) For each $L \in STA_\exists(-,e_i(O(n^l)),k)$ there exists $L' \in STA_\exists(-,O(n),k)$ such that : $x \in L$ iff $\exists y : |y| \leq e_i(|x|^l) \wedge (x,y) \in L'$.

$L' = \{(x,y) : |y| \leq e_i(|x|^l) \wedge x \in L\}$ or $\{xc^m : x \in L \wedge |xc^m| = e_i(|x|^l)\}$ will do the job .

ERUD is contained in $\widetilde{E}$ since $\widetilde{E}$ contains Con and has the necessary closure properties . To prove the converse note that ERUD as well as $\widetilde{E}$ are closed under length bounded quantification where any e_i is used as a length bound . Then the inclusion $\widetilde{E} \subseteq ERUD$ follows by an application of (3) because of $LH \subseteq LSPACE \subseteq \widetilde{E}$. This proves $ERUD = \widetilde{E}$. To prove the equality $ERUD^+ = \widetilde{E}$ it suffices to show $DTIME(e_i(O(n^l))) \subseteq ERUD^+$ because of $U\langle AE_O^{(i)} : i \in N\rangle = \widetilde{E} = ERUD$. However , for each L in $DTIME(e_i(O(n^l)))$ there exists L' in LOGSPACE such that : $x \in L$ iff $\exists y : |y| \leq e_i(O(n^l)) \wedge (x,y) \in L'$. $(x,y) \in L'$ states that y is an accepting computation sequence with input x . This proves (4) . (5) follows from (1) and the well known fact $AE_O^{(i)} \neq AE_O^{(i+1)}$.

It should be mentioned that the representation result in (3) can be used to lift equalities between complexity classes at the linear time level to higher levels, e.g. $LH = LSPACE$ implies $EH^{(i)} = U\langle DSPACE(e_i(O(n^l))) : l \in N\rangle$.

7. Two logspace hierarchies:

In [5] Chandra , Kozen and Stockmeyer considered indexing ATM's , a variant of the ATM's which permits the use of sublinear time bounds . An indexing ATM has an index tape whose content may be interpreted as position of the input which can be accessed. Let $e_{-1}(n)$ be $\log(n)$. The two logspace hierarchies defined below might both be considered as step -1 of the chain of hierarchies discussed earlier . The first hierarchy was introduced in [5] .

$\overline{AEH}^{(-1)} = U\langle \overline{AE}_k^{(-1)} : k \in N\rangle$, $\overline{AE}_O^{(-1)} = LOGSPACE$, $\overline{AE}_k^{(-1)} = U\langle STA_\exists(\log(n^i),-,k) : i \in N\rangle$

$AEH^{(-1)} = U\langle AE_k^{(-1)} : k \in N\rangle$, $AE_O^{(-1)} = LOGTIME$, $AE_k^{(-1)} = U\langle STA_\exists(-,\log(n^i),k) : i \in N\rangle$

We obtain another description of these logspace hierarchies if we replace in the definition of PH the bounds $O(n^l)$ by $\log(n^l)$ and PTIME by LOGSPACE resp. LOGTIME. This yields the hierarchies $\overline{EH}^{(-1)} = \cup\langle\overline{E\Sigma}_k^{(-1)} : k \in N\rangle$ and $EH^{(-1)} = \cup\langle E\Sigma_k^{(-1)} : k \in N\rangle$.

The following proposition shows that $\overline{AEH}^{(-1)}$ is contained in the class $RUD = LH$ whereas $AEH^{(-1)}$ contains the class RUD_{log} of Jones [8] :

Prop.5: (1) $RUD_s \subseteq LOGSPACE \subseteq \overline{AEH}_1^{(-1)} \subseteq RUD^+$, $\overline{AEH}^{(-1)} \subseteq RUD$

(2) $\overline{AE}_k^{(-1)} = \overline{E\Sigma}_k^{(-1)}$ for k in N , $\overline{AEH}^{(-1)} = \overline{EH}^{(-1)}$

(3) $RUD_{log} \subseteq AEH^{(-1)} \subseteq LOGSPACE$

(4) $AE_k^{(-1)} = E\Sigma_k^{(-1)}$ for k in N , $AEH^{(-1)} = EH^{(-1)}$

(1) was proved in Volger [23] . (3) follows since $AEH^{(-1)}$ has the closure properties of RUD_{log} . (2) and (4) can be proved as (1) resp. (2) in proposition 2 .

8. The theories of bounded concatenation:

The question whether linear alternation is more powerful than constant alternation , i.e. whether the inclusions $CATIME(e_i) \subseteq LATIME(e_i)$ and $EH^{(i)} \subseteq LA_i$ are proper , remains open . The classes $LA_i = \cup STA\ (\ ,e_i(O(n^l)),n) : l \in N\rangle$ are closely related to the theories of bounded concatenation . They were introduced by A.R.Meyer in 1975 (cf. [22]) as a uniform method for proving lower bounds for the complexity of first order theories .

The t-bounded concatenation relation Con_t for a given function $t:N \to N$ is defined as follows : $(u,v,w,x) \in Con_t$ iff $uv = w \wedge |w| \leq t(|x|)$. $BCT(\{1,2\}|t)$, the theory of t-bounded concatenation , is the theory $Th((\{1,2\}^*, Con_t, 1, 2))$. Viewed in this context the equality $AEH^{(i)} = EH^{(i)}$ implies that each L in $AEH^{(i)}$ is first order definable in the structure $(\{1,2\}^*, Con_t, 1, 2)$. Recently , we have proved a completeness result for the classes LA_i which in some sense measures the power of bounded concatenation (cf.[22]) .

Prop.6: (1) for all L in $EH^{(i)}$ there is a uniform polynomial time reduction to the decision problem of $BCT(\{1,2\}|e_i)$.

(2) For each L in LA_i there is a polynomial time reduction to the decision problem of $BCT(\{1,2\}|e_i)$.

(3) The decision problem of $BCT(\{1,2\}|e_i)$ belongs to LA_i , whenever $i \geq 1$. In the case $i = 0$ i.e. $LA_o = ATIME(O(n))$ the problem remains open .

9. Conclusion:

The results presented in this paper show that the bounded concatenation relation as well as the different classes of rudimentary languages which are based on it play an important role in that part of complexity theory concerned with the classes LOGSPACE , PTIME , NPTIME etc. . There is also a close connection with time classes with constant resp. linear alternation which should be studied in more detail.

10. References:

[1] Bennett,J.H.: On spectra, Ph.D.Thesis, Princeton Univ., Princeton N.J. 1962, 135 pp.

[2] Berman,L.: The complexity of logical theories, Theoret.Comp.Sci.11(1980), 71-77

[3] Book,R.,Greibach,S.: Quasirealtime languages, Math.Systems Theory 4(1970), 97-111

[4] Chandra,A.K.,Stockmeyer,L.J.: Alternation, in: Proc.17th IEEE Symp.Found. of Comp.Sci.(1976), 98-108

[5] Chandra,A.K.,Kozen,D.C.,Stockmeyer,L.J.: Alternation, J.ACM 28(1981), 114-133

[6] Harrow,K.: The bounded arithmetic hierarchy, Information and Control 36(1978), 102-117

[7] Jones,N.D.: Context-free languages and rudimentary attributes, Math.Systems Theory 3(1969), 102-109, 11(1977/8), 379-380

[8] Jones,N.D.: Space-bounded reducibility among combinatorial problems, J.Comp. System Sci.11(1975), 68-85, 15(1977), 241

[9] King,K.N.,Wrathall,C.: Stack languages and log n space, J.Comp.System Sci.17 (1978), 281-299

[10] Kozen,D.C.: On parallelism in Turing machines, in: Proc.17th IEEE Symp.Found. of Comp.Sci.(1976), 89-97

[11] Meloul,J.: Rudimentary predicates, low level complexity classes and related automata, Ph.D.Thesis, Oxford Univ., Oxford 1979, 210 pp.

[12] Meyer,A.R.,Stockmeyer,L.J.: The equivalence problem for regular expressions with squaring requires exponential space, in: Proc.13th IEEE Symp. Switching and Automata Theory (1972), 125-129

[13] Nepomnjascii,V.A.: Rudimentary predicates and Turing computations, Soviet Math.Dokl.11(1970), 1462-1465

[14] Nepomnjascii,V.A.: Rudimentary interpretation of two-tape Turing computations, Kibernetika (1970) 2, 29-35

[15] Nepomnjascii,V.A.: Examples of predicates not expressible by S-Rud formulae, Kibernetika (1978) 2, 44-46

[16] Orponen,P.: Complexity classes of alternating machines with oracles, in: Proc. 10th Coll. Automata,Languages and Programming (1983), Lecture Notes in Comp. Sci.154, Springer Verlag 1983, 573-584

[17] Quine,W.V.: Concatenation as a basis for arithmetic, J.Symb.Logic 11(1946), 105-114

[18] Simon,J.: Polynomially bounded quantification over higher types and a new hierarchy of the elementary sets, in: Non-classical Logic, Model Theory and Computability, North-Holland Publ.Comp.1977, 267-281

[19] Smullyan,R.: Theory of formal systems, Annals of Math.Studies 47, Princeton Univ.Press 1961, 147 pp.

[20] Stockmeyer,L.J.: The polynomial-time hierarchy, IBM Res.Report RC5379(1975)

[21] Stockmeyer,L.J.: The polynomial-time hierarchy, Theoret.Comp.Sci.3(1977), 1-22

[22] Volger,H.: Turing machines with linear alternation, theories of bounded concatenation and the decision problem of first order theories, Theoret.Comp.Sci. 23(1983), 333-338

[23] Volger,H.: Rudimentary relations and Turing machines with linear alternation, to appear in: Proc.Conf.Recursive Combinatorics, Münster 1983, 6 pp.

[24] Wilkie,A.J.: Applications of complexity theory to Σ_0-definability problems in arithmetic, in: Model Theory of Algebra and Arithmetic, Lecture Notes in Math. 834, Springer Verlag 1980, 363-369

[25] Wilkie,A.J.: On core structures for Peano arithmetic, in: Logic Coll.'80, North-Holland Publ.Comp. 1982, 311-314

[26] Wilkie,A.J.,Paris,J.B.: Models of arithmetic and the rudimentary sets, Bull. Math.Soc.Belg.Sér.B 33(1981), 157-169

[27] Wrathall,C.: Subrecursive predicates and automata, Ph.D.Thesis, Harvard Univ., Cambridge Mass. 1975, 156 pp.
[28] Wrathall,C.: Complete sets and the polynomial-time hierarchy, Theoret.Comp. Sci.3(1977),23-33
[29] Wrathall,C.: Rudimentary predicates and relative computation, SIAM J.Computing 7(1978), 194-209
[30] Yu,Y.Y.: Rudimentary relations and formal languages, Ph.D.Thesis, Univ. of California,Berkeley Cal. 1970, 47 pp.
[31] Yu,Y.Y.: Rudimentary relations and stack languages, Math.Systems Theory 10 (1977), 337-343

Hugo Volger
Mathematisches Institut
Universität Tübingen
Auf den Morgenstelle 10
D 7400 TÜBINGEN