RUDIMENTARY RELATIONS AND TURING MACHINES WITH LINEAR ALTERNATION

Hugo Volger
Math.Institut,Universität Tübingen
D7400 Tübingen, Fed.Rep.of Germany

Abstract: Smullyan [61] introduced the class R of rudimentary relations as the smallest class which contains the concatenation relation and which is closed under the boolean operations, explicit transformations and linearly bounded quantification. RUD, the class of rudimentary languages, consists of the sequential encodings of rudimentary relations. Wrathall [75] has shown that RUD can be described as the union LH of a linear time analogue of the polynomial time hierarchy of PH of Meyer, Stockmeyer [72].

Chandra, Kozen and Stockmeyer [76] introduced the concept of alternating Turing machines (=ATM). An ATM is a nondeterministic Turing machine with two disjoint sets of states, existential and universal states, which play dual roles in the definition of acceptance. A language L belongs to the alternation class STA(s,t,a) if there exists an ATM M such that for each word w in L there exists a finite accepting computation tree of M for w of depth $\leq t(n)$, alternation depth $\leq a(n)$ and space $\leq s(n)$, where $n = |w|$.

There is a close connection between quantification and linear alternation. Chandra, Kozen and Stockmeyer noted that PH may be described as the union of a hierarchy of bounded alternation. An analogous result will be shown for RUD = LH:

(1) $\mathrm{RUD} = \bigcup\langle \mathrm{STA}(-,O(n),k) : k \in N\rangle \subseteq \mathrm{ATIME}(O(n)) = \mathrm{STA}(-,O(n),O(n))$

Extending a result of Nepomnjascii [70] for NLOGSPACE we are able to prove that the alternating LOGSPACE hierarchy of Chandra, Kozen and Stockmeyer is contained in RUD:

(2) $\bigcup\langle \mathrm{STA}(O(n^{\alpha}),O(n^{\beta}),k) : k \in N\rangle \subseteq \mathrm{RUD}$ for $\alpha < 1 \leq \beta$, and hence

(3) $\bigcup\langle \mathrm{STA}(\log(n),-,k) : k \in N\rangle \subseteq \mathrm{RUD}$

However, the question whether the inclusion $\mathrm{RUD} \subseteq \mathrm{ATIME}(O(n))$ is proper remains open. A negative answer would solve important open problems in complexity theory.

1. Rudimentary relations

The class R of rudimentary relations on $\{1,2\}^*$ which is based on the concatenation relation was introduced by Smullyan [61] as a string analogue of the class CA of constructive arithmetic relations on N which is based on + and × . Smullyan showed that R forms a base for the recursively enumerable relations i.e. a relation $Q(\overline{x})$ is r.e. iff there exists a (strictly) rudimentary relation $P(\overline{x},y)$ such that $Q(x) \Leftrightarrow \exists y P(\overline{x},y)$. Moreover, Bennett [62] has shown in his thesis that R coincides with CA via the dyadic coding of natural numbers. In the following we shall identify a word in $\{1,\dots,k\}^*$ with the corresponding natural number under the k-adic coding.

By means of the sequential encoding of relations one obtains the class RUD of <u>rudimentary languages</u> ($\sigma(Q) = \{w_1cw_2c\dots cw_k : (w_1,\dots,w_k) \in Q\}$, where c is new). It has been studied by several authors (Yu [70,77], Jones [69,75], Nepomnjascii [70,72,78], Wrathall [75,78], Meloul [79]).

The class R (resp.XR) of <u>rudimentary</u> (resp. <u>extended rudimentary</u>) <u>relations</u> is the least class which contains the concatenation relation $x \cdot y = z$ and which is closed under the boolean operations, explicit transformations and linearly (resp. polynomially) bounded quantification. The class R^+ (resp.XR^+) of <u>positive</u> (resp. <u>extended positive</u>) <u>rudimentary relations</u> is the least class which contains the concatenation relation and which is closed under union and intersection, explicit transformations, universal subword quantification and linearly (resp. polynomially) bounded existential quantification. The class R_s of <u>strictly rudimentary relations</u> is the least class which contains the concatenation relation and which is closed under the boolean operations, explicit transformations and subword quantification.

$\exists y : y \subseteq x \wedge \dots$	subword quantification
$\exists y : \lvert y\rvert \leq k\lvert x\rvert \wedge \dots$	linearly bounded quantification
$\exists y : \lvert y\rvert \leq \lvert x\rvert^k \wedge \dots$	polynomially bounded quantification

and analogously for universal quantification. It should be noted that $|y| \leq k|x|$ resp. $|y| \leq |x|^k$ may be replaced by $y \leq x^k$ resp. $y \leq x^{(\log x)^{k+1}}$.

Using the sequential encoding of relations we obtain the corresponding classes of languages : RUD, XRUD, RUD^+, $XRUD^+$, RUD_s. Clearly, these classes are related as follows :

$$\begin{array}{ccccc} & & XRUD^+ & \subseteq & XRUD \\ & & \cup\!\mid & & \cup\!\mid \\ RUD_s & \subseteq & RUD^+ & \subseteq & RUD \end{array}$$

The class RUD_s is quite small, since $RUD_s \subseteq$ LOGSPACE and $\{1^n2^n : n \in N\} \not\subseteq RUD_s$ (cf.Nepomnjascii [78]). However, the NP-complete problem SAT is of the form $\exists y : |y| < |x| \wedge Q(y,x)$ with $Q(y,x)$ strictly rudimentary, as Meloul [79] has shown.

2.<u>Turing machines with linear alternation</u>

Chandra and Stockmeyer [76] and Kozen [76] have extended the concept of nondeterministic Turing machines (NTM's) to <u>alternating Turing machines</u> (ATM's). An ATM M is a NTM which has two disjoint sets of states, the existential and universal states , and a distinguished accepting resp. rejecting state . Configurations and successor configurations are defined as for NTM's . An input w is accepted by M (i.e. $w \in L(M)$), if there exists a finite accepting subtree B of the computation tree of 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 C is a node of B labeled with an existential (resp. universal) configuration then at least one (resp. all) successor configuration(s) must appear as labels of the successors of C in B (cf.Chandra , Kozen and Stockmeyer [81]). Note that a NTM is just an ATM with no universal states.

A language L belongs to the alternation class STA(s,t,a) , if L is accepted by an ATM M such that each w in L posesses an accepting subtree B of depth $\leq t(n)$ and alternation depth $\leq a(n)$ and each configuration in B uses at most s(n) cells , where $n = |w|$. In particular , one defines ATIME(t) = STA(-,t,-) and ASPACE(s) = STA(s,-,-) .

Alternating time bridges the gap between nondeterministic time and deterministic space (cf.Chandra,Kozen and Stockmeyer [81]):

$\mathrm{NTIME}(t) \subseteq \mathrm{ATIME}(t) \subseteq \mathrm{DSPACE}(t)$ and APTIME = PSPACE .

Moreover , there is a close connection between alternation and quantification . In particular , hierarchies defined by ressource bounded quantification are closely related to hierarchies defined by bounded alternation classes using the same ressources.

3.The linear - and the polynomial time hierarchy

In her thesis Wrathall [75] has shown that the class XRUD is the union of the polynomial time hierarchy of Meyer , Stockmeyer [72] and that the class RUD is the union of a linear time analogue of this hierarchy. There are three different ways of describing the hierarchy for PH resp. LH :

Nondeterministic Oracles:

$NP_* = \bigcup\langle NP_k : k \in N\rangle$, $NP_0 = PTIME$, $NP_{k+1} = \underline{NP}(NP_k)$

$NL_* = \bigcup\langle NL_k : k \in N\rangle$, $NL_0 = LTIME$, $NL_{k+1} = \underline{NL}(NL_k)$,

where $\underline{NP}(\underline{A})$ resp. $\underline{NL}(\underline{A})$ is the class of languages accepted by a nondeterministic oracle TM (=NOTM) with a polynomial resp. linear time bound and an oracle for a member of the class $\underline{A}$.

Bounded Quantifiers:

$PH = \bigcup\langle P\Sigma_k : k \in N\rangle$, $P\Sigma_0 = PTIME$, $P\Sigma_{k+1} = \exists_p(P\Sigma_k \vee coP\Sigma_k)$

$LH = \bigcup\langle L\Sigma_k : k \in N\rangle$, $L\Sigma_0 = LTIME$, $L\Sigma_{k+1} = \exists_l(L\Sigma_k \vee coL\Sigma_k)$,

where L belongs to $\exists_p(\underline{A})$ resp. $\exists_l(\underline{A})$ if there exists L' in $\underline{A}$ such that x is in L iff $\exists y: |y| \leq |x|^k \wedge L'(x,y)$ resp. $\exists y: |y| \leq k|x| \wedge L'(x,y)$.

Bounded Alternation:

$APH = \bigcup\langle AP_k : k \in N\rangle$, $AP_k = \bigcup\langle STA(-,O(n^i),k) : i \in N\rangle$

$ALH = \bigcup\langle AL_k : k \in N\rangle$, $AL_k = STA(-,O(n),k)$.

The following proposition shows that the three hierarchies coincide in both cases :

Prop.1: (a_1) $NP_k = P\Sigma_k$, $NP_* = PH$, (a_2) $NL_k = L\Sigma_k$, $NL_* = LH$, (b_1) $PH = XRUD$, (b_2) $NP_1 = XRUD^+$, (c_1) $LH = RUD$, (c_2) $NL_1 \subseteq RUD^+$, (d_1) $P\Sigma_k = AP_k$, $PH = APH$, (d_2) $L\Sigma_k = AL_k$, $LH = ALH$.

(a_1) - (c_1) can be found in Wrathall [77,78]. (c_2) follows from the inclusion $CFL \subseteq RUD$ in Yu [70] by means of a result of Book,Greibach [73]. (d_1) was mentioned in Chandra,Kozen and Stockmeyer [81] , but (d_2) seems to be new .

(d_1) and (d_2) can be proved by the same method . Given the syntactical description of L with at most k alternations of bounded quantifiers it is easy to construct an ATM accepting L with the corresponding time bound and at most k alternations . Conversely , given an ATM accepting a language L with at most k alternations , one constructs a DTM accepting a language L' and having k additional tapes , one for each alternation phase . Being in the i-th alternation phase the machine makes use of the content of the i-th tape going from left to right to control the choice of moves as long as necessary and then changes to the (i+1)-th tape if an alternation occurs. Hence L can be obtained from L' by appropriate bounded quantification as desired . This should be compared with the incremental stack automata in Yu [70] .

Meloul [79] presented two different quantifier hierarchies for PH and LH which are based on LOGSPACE in both cases rather than PTIME and LTIME . In addition , he exhibited corresponding oracle hierarchies . He worked with LOGSPACE oracle machines which have a choice tape as source of nondeterminism and required that the length of the choice tape and the oracle tape is polynomially resp. linearly bounded .

The classes XRUD and RUD are related to alternation classes with a linear amount of alternation as follows :

$APH \subseteq LAPTIME = \bigcup\langle STA(-,O(n^i),O(n)) : i \in N\rangle \subseteq PSPACE$

$ALH \subseteq ALTIME = STA(-,O(n),O(n)) \subseteq LSPACE$

It should be noted that the PSPACE-complete problem QBF is in ALTIME . The question whether these inclusions are proper remains open. A new quantifier seems to be involved . A positive answer would resolve important open problems in complexity theory .

It is known that questions concerning the polynomial time hierarchy may be reduced to questions concerning the linear time hierarchy with the help of polynomial padding. A language L is replaced by the padded language $L_k = \{wc^m : m = |w|^k , w \in L\}$ or $L^k = \{(x,y) : |y| = |x|^k, x \in L\}$. Making use of the fact that SAT and the relation $|y| = |x|^k$ are positive rudimentary (cf. Meloul [79]) one arrives at the following situation :

Prop.2: (i) LTIME = RUD^+ implies $RUD^+ \subseteq$ PTIME (iff PTIME = $XRUD^+$)
(ii) RUD^+ = RUD implies RUD $\subseteq XRUD^+$ (iff $XRUD^+$ = XRUD)
(iii) RUD = ALTIME implies ALTIME $\subseteq$ XRUD (iff XRUD = LAPTIME)
(iv) ALTIME = LSPACE implies LSPACE $\subseteq$ LAPTIME (iff LAPTIME = PSPACE)

4.Alternating LOGSPACE hierarchy

The alternating LOGSPACE hierarchy was defined by Chandra, Kozen and Stockmeyer [81] as follows: ALOGH = $\cup\langle ALOG_k : k \in N\rangle$ with STA(log n,-,k) = $ALOG_k$. In addition, they have shown : NLOGSPACE $\subseteq$ ALOGH $\subseteq$ DSPACE($\log^2 n$), ALOGSPACE = PTIME. Extending the result of Nepomnjascii [70] we are able to show the following :

Prop.3: (*) NLOGSPACE $\subseteq$ NST($O(n^\alpha)$,$O(n^\beta)$) $\subseteq RUD^+$ for $\alpha < 1 \leq \beta$
(**) STA(log n,-,k) $\subseteq$ STA($O(n^\alpha)$,$O(n^\beta)$,k) $\subseteq$ RUD for all k and $\alpha < 1 \leq \beta$
(***) ALOGH $\subseteq$ RUD .

Nepomnjascii exhibited a binary relation in R^+ which expresses the existence of a sequence of $n^{1-\alpha}$ successive configurations of length n^α. Having iterated this process $t = \lfloor \beta/1-\alpha \rfloor$ times, one is able to describe n^β successive configurations of length n^α. A final existential quantifier with a linear bound yields (*). Now we adapt the method in Berman [80] of expressing acceptance in ATM's with at most k alternations. This introduces k additional quantifiers with linear bounds and yields (**). (***) is an immediate consequence of (**).

Yu [77] has shown that the class 1NSA of languages accepted by 1-way nondeterministic stack automata is contained in RUD. Using the same method as above this result can be extended as follows. The class $\cup\langle 1ASA_k : k \in N\rangle$ is contained in RUD, where $1ASA_k$ is the class of languages accepted by 1-way alternating stack automata with at most k alternations.

References:

Bennett,J.H.[62]: On Spectra, Ph.D.Thesis, Princeton Univ., Princeton N.J., 1962, 135pp.
Berman,L.[80]: The Complexity of Logical Theories, Theoret.Comp.Sci.11 (1980),71-77
Book,R.,Greibach,S.[70]: Quasirealtime Languages, Math.Systems Theory 4 (1970),97-111
Chandra,A.K.,Stockmeyer,L.J.[76]: Alternation, in: Proc.17th IEEE Symp. on Found.of Comp.Sci.(1976),98-108
Chandra,A.K.,Kozen,D.C.,Stockmeyer,L.J.[81]: Alternation,J.ACM 28(1981), 114-133

Harrow,K.[78]: The Bounded Arithmetic Hierarchy, Information and Control 36(1978),102-117
Jones,N.D.[69]: Context-free Languages and Rudimentary Attributes, Math. Systems Theory 3(1969)102-109, 11(1977/8),379-380
Jones,N.D.[75]: Space-Bounded Reducibility among Combinatorial Problems, J.Comp.System Sci.11(1975),68-85, 15(1977),241
King,K.N.,Wrathall,C.[78]: Stack Languages and Log n Space, J.Comp.System Sci.17(1978),281-299
Kozen,D.C.[76]: On Parallelism in Turing Machines, in: Proc.17th IEEE Symp. on Found.of Comp.Sci.(1976),89-97
Meloul,J.[79]: Rudimentary Predicates, Low Level Complexity Classes and Related Automata, Ph.D.Thesis, Oxford Univ., Oxford 1979, 210pp.
Meyer,A.R.,Stockmeyer,L.J.[72]: The Equivalence Problem for Regular Expressions with Squaring requires Exponential Space, in: Proc.13th IEEE on Switching and Automata Theory(1972),125-129
Nepomnjascii,V.A.[70a]: Rudimentary Predicates and Turing Computations, Soviet Math.Dokl.11(1970),1462-1465
Nepomnjascii,V.A.[70b]: Rudimentary Interpretation of Two-Tape Turing Computation, Kibernetika 6(1970),29-35
Nepomnjascii,V.A.[78]: Examples of Predicates not expressible by S-RUD Formulae, Kibernetika 12(1978)
Ruzzo,W.L.[80]: Tree-Size Bounded Alternation, J.Comp.System Sci.21 (1980),218-235
Ruzzo,W.L.,Simon,J.,Tompa,M.[82]: Space-Bounded Hierarchies and Probabilistic Computations, in: Proc.14th ACM Symp. on Theory of Computing, San Francisco 1982,215-223
Smullyan,R.[61]: Theory of Formal Systems, Annals of Math.Studies 47, Princeton Univ.Press 1961, 147pp.
Stockmeyer,L.J.[75]: The Polynomial-Time Hierarchy, IBM Fesearch Report RC 5379(1975)
Stockmeyer,L.J.[77]: The Polynomial-Time Hierarchy, Theoret.Comp.Sci.3 (1977),1-22
Volger,H.[83]: Turing Machines with Linear Alternation,Theories of Bounded Concatenation and the Decision Problem of First Order Theories, Theoret. Comp.Sci.23(1983),333-338
Wrathall,C.[75]: Subrecursive Predicates and Automata, Ph.D.Thesis, Harvard Univ.,Cambridge MA,1975, 156pp.
Wrathall,C.[77]: Complete Sets and the Polynomial-Time Hierarchy,Theoret. Comp.Sci.3(1977),23-33
Wrathall,C.[78]:Rudimentary Predicates and Relative Computation, SIAM J. Computing 7(1978),194-209
Yu,Y.Y.[70]: Rudimentary Relations and Formal Languages, Ph.D.Thesis, Univ.of California,Berkeley 1970, 47pp.
Yu,Y.Y.[77]: Rudimentary Relations and Stack Languages, Math.Systems Theory 10(1977),337-343