ELSEVIER

Annals of Pure and Applied Logic 78 (1996) 189–202

ANNALS OF PURE AND APPLIED LOGIC

Arity and alternation in second-order logic

J.A. Makowsky[a,*,1], Y.B. Pnueli[b,2]

[a] *Department of Computer Science, Technion – Israel Institute of Technology, Haifa, Israel*
[b] *Institut für Informatik, Freie Universität Berlin, Berlin, Germany*

Received 1 November 1994; communicated by A. Nerode

Abstract

We investigate the expressive power of second-order logic over finite structures, when two limitations are imposed. Let $SAA(k, n)$ ($AA(k, n)$) be the set of second-order formulas such that the arity of the relation variables is bounded by k and the number of alternations of (both first-order and) second-order quantification is bounded by n. We show that this imposes a proper hierarchy on second-order logic, i.e. for every k, n there are problems not definable in $AA(k, n)$ but definable in $AA(k + c_1, n + d_1)$ for some c_1, d_1.

The method to show this is to introduce the set $AUTOSAT(F)$ of formulas in F which satisfy themselves. We study the complexity of this set for various fragments of second-order logic. For first-order logic FOL with unbounded alternation of quantifiers $AUTOSAT(FOL)$ is *PSpace*-complete. For first-order logic FOL_n with alternation of quantifiers bounded by n, $AUTOSAT(FOL_n)$ is definable in $AA(3, n + 4)$. $AUTOSAT(AA(k, n))$ is definable in $AA(k + c_1, n + d_1)$ for some c_1, d_1.

1. Introduction

In this paper we deal with second-order logic over finite structures. Let τ be a first-order vocabulary. We denote by $SOL(\tau)$ ($FOL(\tau)$) the set of second-order (first-order) formulas over τ. Clearly, $FOL(\tau) \subseteq SOL(\tau)$. For $\phi \in SOL(\tau)$ we denote by $Mod(\phi)$ the class of finite τ-structures $\mathscr{A}$ such that $\mathscr{A} \models \phi$. Let $F = F(\tau) \subseteq SOL(\tau)$. A class of finite (ordered) τ–structures K is F-definable if it is of the form $Mod(\phi)$ for some $\phi \in F$. It is well known, [10], that K is SOL-definable iff K is in the polynomial hierarchy PH. We investigate the expressive power of second-order logic over finite structures, when two limitations are imposed: on the number of alternations of

* Corresponding author. E-mail: janos@cs.technion.ac.il.

[1] Partially supported by a Grant of the French–Israeli Binational Foundation and by the Fund for Promotion of Research of the Technion – Israeli Institute of Technology.

[2] Since Fall 1993 Minerva Fellow.

SSDI 0168-0072(95)00013-5

quantifiers and on the arity of the second-order variables. A preliminary version of our results has appeared as [12].

Definition 1. A first- (second-) order formula ϕ is in *prenex normal form* if it is of the form

$$Q_1 V_1 Q_2 V_2, \dots, Q_m V_m B,$$

where V_i is a first-order (either a first-order or a second-order) variable and Q_i is either $\exists$ or $\forall$ and B is quantifier free.

(ii) A second-order formula ϕ is in *doubly prenex normal form* if it is in prenex normal form of the form

$$Q_1 V_1 Q_2 V_2, \dots, Q_l V_l q_1 v_1 q_2 v_2, \dots, q_m v_m B,$$

where V_i is a second-order variable, v_i is a first-order variable and Q_i and q_i are either $\exists$ or $\forall$.

It is well known that every second- (first-) order formula is equivalent to a second- (first-) order formula in prenex normal form. If we disregard the arity of the second-order variables it is also equivalent to a second-order formula in doubly prenex normal form. Also, every first-order formula is equivalent to an existential second-order formula where the second-order quantification is followed by at most one alternation of first-order quantifiers. That arity plays an essential rôle here is shown by the following observation, due to R. Fagin:[3]

Proposition 2 (Fagin). *Let τ be the vocabulary of graphs. We consider here undirected graphs only. There is a formula*

$$\phi = \forall \bar{x} \exists \bar{U} \theta$$

such that $\bar{U}$ is a vector of unary relation variables and θ is first-order, and such that ϕ is not equivalent to any formula

$$\psi = \exists U_1 \dots \exists U_n \chi$$

with U_i unary and χ first-order.

Proof. By a result of Kannelakis, cf. [1], there is a formula

$$\phi_1(x, y) = \exists \bar{U} \theta(x, y)$$

with $\bar{U}$ a vector of unary relation variables, which says that there is a path from x to y. So $\phi = \forall x, y\, \phi_1(x, y)$ says that a graph is connected. But by the result of Ajtai and Fagin [1], connectivity is not expressible by an existential monadic second-order formula. □

This leads us to the following definitions of hierarchies which incorporate both alternation of quantification and arity of the second-order variables.

[3] We are indebted to R. Fagin, who allowed us to include this result.

We define now hierarchies $AA\Sigma(k,n)$, $AA\Pi(k,n)$, $AA(k,n)$, the *strict AA-Hierarchy* (*alternation-arity hierarchy*) and $SAA\Sigma(k,n)$,$SAA\Pi(k,n)$, $SAA(k,n)$, the *second-order AA-Hierarchy* (*alternation-arity hierarchy*) inductively as follows:

Definition 3 (*AA-Hierarchy*). (i) $AA\Sigma(0,0) = AA\Pi(0,0) = AA(0,0)$ consists of the quantifier-free first-order formulas.

(ii) $AA\Sigma(k,0) = AA\Pi(k,0) = AA(k,0)$ consists of the quantifier-free second-order formulas, where each relation variable is of arity $m \leqslant k \in \mathbf{N}$.

(iii) $AA\Sigma(k, n+1)(AA\Pi(k, n+1))$ consists of the formulas of the form

$$Q_1 V_1 Q_2 V_2, \ldots, Q_m V_m \phi,$$

where V_i is either a first-order or a second-order variable (of arity $\leqslant k$) and each Q_i is $\exists$ ($\forall$) and $\phi \in AA\Pi(k,n)(\phi \in AA\Sigma(k,n))$.

(iv) $AA(k,n) = AA\Sigma(k,n) \cup AA\Pi(k,n)$.

(v) The *FA-Hierarchies* (*first-order alternation hierarchy*) $FA\Sigma(n)$, $FA\Pi(n)$, $FA(n)$ are defined by restricting formulas of the strict AA-hierarchy to first-order variables only.

Definition 4 (SAA-Hierarchy). (i) $SAA\Sigma(k,0) = SAA\Pi(k,0) = SAA(k,0)$ consists of the second-order formulas in prenex normal form with all the quantifiers first-order and where each relation variable is of arity $m \leqslant k \in \mathbf{N}$.

(ii) $\mathrm{SAA}\Sigma(k, n+1)(SAA\Pi(k, n+1))$ consists of the formulas of the form

$$Q_1 V_1 Q_2 V_2, \ldots, Q_m V_m \phi,$$

where V_i is a second-order variable (of arity $\leqslant k$) and each Q_i is $\exists$ ($\forall$) and $\phi \in SAA\Pi(k,n)(\phi \in SAA\Sigma(k,n))$.

(iii) $SAA(k,n) = SAA\Sigma(k,n) \cup SAA\Pi(k,n)$.

Remark. In the AA-hierarchies we count first- and second-order quantifiers wherever they occur. In the SAA-hierarchies we require that the second-order quantifiers precede the first-order quantifiers and we then disregard first-order quantifiers.

From this we gather

Fact 5. (i) *The usual operations on renaming variables and the prenex normal forms gives that both* $FA(n)$, $AA(k,n)$ and $SAA(k,n)$ *are closed under boolean operations.*

(ii) *Using Skolem Normal Forms we get* $FOL \subseteq \bigcup_k AA\Sigma(k,3)$.

(iii) *From the previous we also get* $\bigcup_k SAA(k,n) \subseteq \bigcup_k AA(k, n+3)$.

(iv) $\bigcup_n SAA(k,n) \subseteq \bigcup_n AA(k,n)$.

Our main result shows that this imposes a proper hierarchy on second-order logic, i.e.

Theorem A. *For every $k, n \in \mathbf{N}$ there are problems not definable in $AA(k, n)$ but definable in $AA(k + c_1, n + d_1)$ for some $c_1, d_1 \in \mathbf{N}$.*

The method to show this is to introduce the set $AUTOSAT(F)$ of formulas in F which satisfy themselves. We study the complexity of $AUTOSAT(F)$ for various fragments F of second-order logic. If we allow unbounded alternation of first-order quantifiers we have:

Theorem B. (i) *$AUTOSAT(FOL)$ is PSpace-complete.*
(ii) *For every $k, n \in \mathbf{N}$ $AUTOSAT(SAA(k, n))$ is PSpace-complete.*

If we bound the number of alternations of all quantifiers we have:

Theorem C. (i) *For every n, $AUTOSAT(FA(n))$ is complete for the class $\Sigma_n^p \cup \Pi_n^p$ of the polynomial hierarchy.*
(ii) *$AUTOSAT(FA(n))$ is definable in $AA(3, n + 4)$.*
(iii) *$AUTOSAT(AA(k, n))$ is definable in $AA(k + c(k), n + 4)$ where $c(k) = 1$ for $k > 1$ and $c(k) = 2$ for $k = 1$.*
(iv) *$AUTOSAT(SOL)$ is in $TIME(2^{n^{\varepsilon}})$ for every $\varepsilon \in \mathbf{R}^+$.*

Theorem A now follows from Theorem C by observing that the complement of $AUTOSAT(F)$ is not definable in F, hence, if F is closed under negation, neither is $AUTOSAT(F)$.

Remark. We should note that recently Grohe [7] has shown that the bounded-arity hierarchies in fixed point logic are strict. In these hierarchies arbitrary quantifier depth and fixed point depth are allowed. It would be natural to conjecture that this is also the case for second-order logic, but our techniques do not give such a result.

2. The Polynomial Hierarchy

We now discuss the relationship of $AA(k, n)$ and $SAA(k, n)$ with various complexity classes in the Polynomial Hierarchy. First we note [9]:

Fact 6 (Stockmeyer and Lynch). *For every level Σ_n^p and Π_n^p of the Polynomial Hierarchy we have*

$$\Sigma_n^p = \bigcup_k SAA\Sigma(k, n), \qquad \Pi_n^p = \bigcup_k SAA\Pi(k, n)$$

and

$$\Sigma_n^p \subseteq \bigcup_k SAA(k, n) \subseteq \Sigma_{n+1}^p.$$

Fact 7 (Stockmeyer and Lynch). *For every level Σ_n^p and Π_n^p of the Polynomial Hierarchy we have*

$$\Sigma_n^p \subseteq \bigcup_k AA\Sigma(k, n+2), \qquad \Pi_n^p \subseteq \bigcup_k AA\Pi(k, n+2).$$

This follows from the fact that in the Stockmeyer and Lynch characterization of Σ_n^p there are n alternations of second-order quantifiers, followed by a first-order formula of the form $\forall \bar{x} \exists \bar{y} \theta$ with θ quantifier free.

Remark. Fact 7 could be used for an alternative way of slicing the polynomial hierarchy. We look only at formulas in doubly prenex normal form where the first-order part is of the form $\forall \bar{x} \exists \bar{y} \theta$ and then count only second order quantification and restrict the arities of the second-order variables. We leave it to the reader to restate our results. From this point of view,

Fact 8 (Lynch [10]). *If K is in* **NP** *and recognizable in $NTIME(n^d)$, then K is definable in $SAA(d, 1)$ (existential) and even $AA(d, c)$ for some $c \in \mathbf{N}$.*

Remark. The converse of Fact 8 is not true. The reason is that even in $AA(d, c)$ we count only blocks of existential (universal) quantifiers. If a converse were provable, our Theorem A would trivialize, by the hierarchy theorem of Seiferas, Fischer and Meyer (Theorem H3 in [9]).

The next facts show that the SAA-hierarchy starts as a proper hierarchy.

Fact 9 (de Rougement [5]). *Connectivity of undirected graphs is in $SAA\Pi(1, 1)$ but not in $SAA\Sigma(1, 1)$* [5].

Fact 10 (Turán [14], de Rougemont [5] and Makowsky [11]). *Let HAM be the class of finite undirected Hamiltonian graphs.*

(i) *HAM is* **NP**-*complete and definable in $SAA(2, 1)$ (existential) and even $AA(2, c)$ for some $c \in \mathbf{N}$.*

(ii) *But HAM is not definable in any $SAA(1, n)$ (and hence in any $AA(1, n)$).*

Remark. Turán proved this first for undirected graphs. De Rougemont only, independently but later, proved that HAM is not definable in $SAA\Sigma(1, 1)$. Makowsky proved it also for undirected graphs with an additional order. His proof generalizes to other classes of graphs, such as ordered graphs with a clique of at least half their size.

The next observation shows that already in monadic second-order logic there are problems of arbitrarily high complexity within the polynomial hierarchy.

Proposition 11 (Folklore). *For every level Σ_n^p of the Polynomial Hierarchy there is a problem K_n* complete for *Σ_n^p (via polynomial reductions) definable in $SAA(1, n)$ and even $AA(1, n + c)$ for some $c \in \mathbf{N}$.*

Proof (*sketch*). Easy, using suitable codings for $QSAT_n$. □

In other words:

Corollary 12. *The polynomial hierarchy collapses to level k iff $SAA(1, n) \subseteq SAA(1, k)$ for every $n \in \mathbf{N}$.*

3. Context-free languages

Here we state and prove some observations which we need later. They concern the complexity of recognizing well formed formulas of first- and second-order logic.

Fact 13 (Büchi [2] and Thomas [13]). *Every regular language L is definable in $SAA\Sigma(1, 1)$ and even $AA\Sigma(1, 2)$.*

Proposition 14. *Every context-free language L is definable in $SAA(3, 1)$ (existential) and even $AA(3, 3)$.*

Proof. We use the fact that every CF-language can be generated by a CF-grammar in Chomsky normal form [8] (for A, B, C variables and a a character of the alphabet, the rules are either of the form $A \to BC$ or $A \to a$). We build for every such grammar a formula which defines the generated language: with each rule i of form $A \to BC$ in the grammar, we associate a ternary relation symbol R_i so that $R_i(x_1, x_2, x_3)$ holds in a word $\mathscr{A}$, if the characters in places x_1 to x_3 evolved from a single symbol A via the application of rule i in such a way that the characters from x_1 to $x_2 - 1$ continued to evolve from the B symbol and x_2 to x_3 continued from the C symbol.

A formula for a grammar with m such rules can be written in the form

$$\exists R_1, R_2, \ldots, R_m \forall x_1, x_2, x_3, x_4, x_5, x_6 \exists x_7, x_8, x_9 \Phi$$

where Φ is quantifier free and says that the R's indeed have the meaning we want them to have.

Specifically Φ is $\phi_{\text{order}} \wedge \phi_{\text{proper}} \wedge \phi_{\text{correct}} \wedge \phi_{\text{start}}$ where the ϕ's are as follows: ϕ_{order} is a disjunction of formulas of the form

$$R_i(x_1, x_2, x_3) \to ((x_1 < x_2) \wedge (x_2 \leqslant x_3)),$$

ϕ_{proper} says that the R's induce a generation tree, and is a disjunction of formulas of the form

$$(R_{i_1}(x_1, x_2, x_3) \wedge R_{i_2}(x_4, x_5, x_6)) \to ((x_3 < x_4) \vee (x_6 < x_1) \vee (x_1 \leqslant x_4 \wedge x_6 < x_2)$$
$$\vee (x_2 \leqslant x_4 \wedge x_6 \leqslant x_3) \vee (x_4 \leqslant x_1 \wedge x_3 < x_5) \vee (x_5 \leqslant x_1 \wedge x_3 \leqslant x_6)),$$

ϕ_{correct} says that the generation rules are indeed those of the specific grammar, and is a disjunction of formulas of the form

$$\begin{aligned} &R_i(x_1, x_2, x_3) \to (x_8 = x_2 - 1) \\ &\quad \wedge (((x_1 = x_8) \wedge (r_{i_1}(x_1) \vee r_{i_2}(x_1) \ldots)) \vee (R_{i_1}(x_1, x_7, x_8) \vee R_{i_2}(x_1, x_7, x_8) \vee \cdots) \\ &\quad \wedge ((x_2 = x_3) \wedge ((r_{i_1}(x_2) \vee r_{i_2}(x_2) \ldots)) \vee (R_{i_1}(x_2, x_7, x_3) \vee R_{i_2}(x_2, x_7, x_3) \vee \ldots) \end{aligned}$$

where the r's and R's depend on the specific rules of the grammar.
ϕ_{start} says that the whole word was generated from the initial symbol, and is of the form

$$x_9 < x_1 \vee x_9 > x_3 \vee R_{i_1}(x_1, x_9, x_3) \vee R_{i_2}(x_1, x_9, x_3) \vee \ldots$$

where again the R's depend on the rules of the grammar. □

From these facts we conclude that

Proposition 15. *The notion of well-formed formulas of FOL* and *SOL are not definable in $SAA\Sigma(1,1)$ but definable in $SAA\Sigma(3,1)$.*

For similar results, cf. [4, 6].

4. Self-satisfying sentences

Let F be a subset of SOL-formulas. In this section we discuss in detail the class of structures $AUTOSAT(F)$.

Let τ_{SOL} be a finite vocabulary rich enough to describe $SOL(\tau)$-formulas for arbitrary τ. τ_{SOL} consists of one binary relation symbol denoted by $<$ and interpreted as a linear order and many unary relation symbols to distinguish parentheses, quantifiers, variables, relation constants from any τ, indices, etc. In short, for every finite τ and $\phi \in SOL(\tau)$, ϕ can be viewed as a τ_{SOL}-structure. For a formula (sentence) $\phi \in SOL(\tau_{SOL})$ we denote by $\mathscr{A}_\varphi$ the τ_{SOL}-structure isomorphic to ϕ.

Definition 16. Let F be a subset of $SOL(\tau_{SOL})$ which is closed under boolean operations.

(i) We denote by $WFF(F)$ the set of finite τ_{SOL}-structures $\mathscr{A}_\varphi$ such that $\phi \in F$.

(ii) We denote by $AUTOSAT(F)$ the set of finite τ_{SOL}-structures $\mathscr{A}_\varphi$ such that $\phi \in F$ and $\mathscr{A}_\varphi \models \phi$.

(iii) We set $Diag(F) = WFF(F) \setminus AUTOSAT(F)$.

Using the Russel Paradox, we observe:

Fact 17. *Let F be a subset of $SOL(\tau_{SOL})$ which is closed under boolean operations and such that $WFF(F)$ is definable in F. Then $Diag(F)$, and hence, $AUTOSAT(F)$, are not definable in F.*

Proposition 18. *$WFF(SOL)$ is definable in $SAA(3, 1)$ and even in $AA(3, 3)$.*

Proof. As the arity is bounded, we choose a vocabulary τ_{SOL} such that second-order variables of different arities have different symbols. Then, the sets $WFF(SAA(k, n))$ and $WFF(AA(k, n))$ are context free for fixed k. If we encode the arity of the relation symbols in unary, the set $WFF(SOL)$ becomes context free. □

Proposition 19. (i) *For every n $AUTOSAT(FA(n))$ is recognizable in $\Sigma_n^p \cup \Pi_n^p$ of the polynomial hierarchy.*

(ii) *$AUTOSAT(FOL)$, $AUTOSAT(SAA(k, n))$ are recognizable in* **PSpace**.

Proof. For $AUTOSAT(FOL)$, let ϕ be some formula in FOL and let ψ denote its quantifier-free part. Clearly $\mathscr{A}_\varphi \models \phi$ can be established by checking the value of the meaning function for every possible substitution of those variables free in ψ. Since the number of free variables in ψ is bounded by n (the size of $\mathscr{A}_\varphi$, the space required for writing a substitution is $O(n \log(n))$. Given a substitution the verification takes no more than an additional $O(\log(n))$ space. Thus $AUTOSAT(FOL)$ can be recognized in space $O(n \log(n))$.

For $AUTOSAT(SAA(k, n))$, let ϕ be some formula in $SAA(k, n)$ and ψ its unquantified part; $\mathscr{A}_\varphi \models \phi$ can be established by checking the value of the meaning function for every possible substitution of the (first + second-order) variables free in ψ. The size of a substitution in this case is bounded by $O(n^k)$ (the worst case being when all the quantified variables are second-order of arity k). Again given a substitution the verification is $O(\log(n))$ and hence $AUTOSAT(SAA(k, n))$ can be recognized in space $O(n^k)$.

For $AUTOSAT(FA(n))$ we use the same technique except that instead of generating all substitutions we nondeterministically guess a correct one. This is done inductively on n: For $\phi \in FA(n)$, removing the outermost quantifer gives a formula $\psi \in FA(n-1)$. Given a substitution for the free variable, $\mathscr{A}_\varphi \models \psi$ can be verified in $\Sigma_{n-1}^P \cup \Pi_{n-1}^P$ (or

LogSpace for $n = 1$). To verify $\phi \in FA(n)$, all that remains is to nondeterministically guess for the free variable a substitution which either satisfies or dissatisfies ψ (depending on whether the outermost quantifier is existential or universal). This can be done in $\Sigma_n^P \cup \Pi_n^P$. □

Proposition 20. (i) $AUTOSAT(FA(n))$ is $\Sigma_n^P \cup \Pi_n^P$ hard.

(ii) $AUTOSAT(FOL)$ *is* **PSpace** *hard.*

(iii) $AUTOSAT(SAA(k, n))$ *is* **PSpace** *hard.*

Proof (*By a reduction from* $QSAT$ *to* $AUTOSAT(FOL)$). Given ϕ, a quantified boolean formula, we produce in polynomial time ψ, a formula in FOL, such that ϕ is true iff ψ satisfies itself. Let i be some number not used as the index of any variable in ϕ and let $r_\exists(x)$ be the relation symbol denoting that x is a character "$\exists$". ψ is then written as $\exists x_i \phi_{x_j}^{r_\exists(x_j)}$ where $\phi_{x_j}^{r_\exists(x_j)}$ is the FOL formula we get by replacing every appearance of each boolean variable x_j by the corresponding subformula $r_\exists(x_j)$. Clearly ψ is constructable in linear time. Also clearly ϕ is true iff ψ is self satisfying. (Note that the first character in ψ is "$\exists$", and the second is not, so we have both the "TRUE" and "FALSE" boolean assignments.)

Since $QSAT$ is **PSpace**-complete this proves (ii). The same reduction, but with a bounded number of quantifier alternations, gives us (i) (as $QSAT_n$ with n the number of quantifier alternations is complete for $\Sigma_n^P \cup \Pi_n^P$). Since for every k and n, every formula in FOL is also a formula of $SAA(k, n)$ we also get (iii). □

These two propositions establish Theorem B and part (i) of Theorem C.

5. Proof of Theorem C (parts (ii) and (iii))

We use a vocabulary τ_{SOL} such that first-order variables have different symbols from second-order variables and also second-order variables of different arities have different symbols (this can be done if the arity is bounded). In $\mathscr{A}_\varphi$ a variable x_i is written as the appropriate variable symbol "x" followed by the index, written in binary and encapsulated in parenthesis (for example R_5^7 – a 7-ary second-order variable with index 5 – is written as $G(101)$ where G is the symbol for 7-ary second-order variables).

Definition 21. (i) $VAR_i(x)$ are predicates indicating that the character in position x of a word is a symbol of a variable of arity i ($i = 0$ being a first-order variable).

(ii) $INDEX_i(x, y_1, y_2)$ are predicates indicating that the character in position x is a variable of arity i and the characters in positions y_1 to y_2 make up its index.

(iii) $SAME_i(x_1, x_2)$ are predicates indicating that the characters in positions x_1 and x_2 are variables of arity i and they have the same index (that is they refer to the *same* variable).

Lemma 22. *VAR_i, $INDEX_i$ and $SAME_i$ can be written as formulas in $FA(0)$, $FA(2)$ and $AA\Sigma(2, 3)$ respectively.*

Proof. $VAR_i(x)$ is simply $r_{\sigma_i}(x)$ where σ_i is the symbol used to indicate variables of arity i.

$INDEX_i(x, y_1, y_2)$ simply says $VAR_i(x)$ and every symbol between y_1 to y_2 is a binary digit and the symbols $y_1 - 1$ and $y_2 + 1$ are "(" and ")" respectively. This can be done in $FA\Pi(2)$.

$SAME(x_1, x_2)$ says that there are elements y_1, y_2, y_3, y_4 and a binary relation R such that $INDEX_i(x_1, y_1, y_2)$ and $INDEX_i(x_2, y_3, y_4)$ and R can be viewed as an order preserving and symbol preserving one-to-one and onto function from the range $y_1 - y_2$ to the range $y_3 - y_4$. This can be done in $AA\Sigma(2, 3)$. □

Next we want to speak of assignments for free variables in a formula. We view these as relations with an arity one higher than that of the variable to which it assigns a value. Thus an assigment for second-order variables of arity i will be a relation of arity $i + 1$. We denote such a relation by Z_i. First order assignments are a special case as they, too, are relations of arity 2 with the additional restriction that each element corresponding to an FOL variable is in relation with exactly one element. (For our purpose Z_i is meaningless for those elements of the formula which are not variable symbols of arity i.) For bounded k we denote by Z a vector $Z_0, Z_1, \ldots, Z_k$ with the Z's as above.

Definition 23. (i) $ASS_i(R)$ are predicates indicating that R, an $i + 1$-ary relation, is an assignment for the free i-ary variables in a formula.

(ii) $ASS(R)$ is a predicate indicating that the relations R together comprise an assignment for all the free variables in a formula.

Lemma 24. *$ASS_i(R)$ and $ASS(R)$ can be written as formulas in $AA\Pi(2, 3)$.*

Proof. For ASS_i with $i \neq 0$ we write

$$\forall x_1, x_2, y \neg SAME(x_1, x_2) \vee (R(x_1, y) \Leftrightarrow (x_2, y)).$$

For ASS_0 we "and" to the above

$$(\forall x_1 \exists x_2 r_x(x_1) \rightarrow R(x_2, x_2)) \wedge (\forall x_3, x_4, x_5(R(x_3, x_4) \wedge R(x_3, x_5)) \rightarrow x_4 = x_5)$$

where "x" is the symbol for first-order variables. $ASS(R)$ is written as the disjunction of the $k + 1$ formulas for ASS_i. □

Definition 25. (i) $WFF(x_1, x_2)$ is a predicate indicating that the characters in positions $x_1 - x_2$ form a well formed subformula.

(ii) $ATOM_{r_i}(x_1, x_2)$ are predicates indicating that the characters in positions $x_1 - x_2$ form an atomic subformula of the form $r_i(x)$.

(iii) $ATOM_{R^i}(x_1, x_2)$ are predicates indicating that the characters in positions $x_1 - x_2$ form an atomic subformula of the form $R^i(x_1, \ldots, x_i)$.

(iv) $POS_i(x_1, x_2)$ are predicates indicating that the character x_1 is a second-order variable of arity at least i and x_2 is the first-order variable in the ith position of it.

(v) $NOT(x_1, x_2)$ is a predicate indicating that the characters in positions $x_1 - x_2$ form a subformula of the form $\neg(\psi)$.

(vi) $AND(x_1, x_2, x_3)$ is a predicate indicating that the characters in positions $x_1 - x_3$ form a subformula of the form $(\psi_1) \wedge (\psi_2)$ with x_2 the position of the $\wedge$ symbol.

(vii) $OR(x_1, x_2, x_3)$ is a predicate indicating that the characters in positions $x_1 - x_3$ form a subformula of the form $(\psi_1) \vee (\psi_2)$ with x_2 the position of the $\vee$ symbol.

Lemma 26. *The above predicates can be written as formulas of $AA(3, 1)$.*

Proof. Follows as they are all context free. □

Definition 27. (i) $ATOMSAT(Z, x_1, x_2)$ is a predicate indicating that Z is an assignment and the characters in positions $x_1 - x_2$ form an atomic subformula, which is satisfied by the Z assignment for a structure which is the whole formula.

(ii) $QFREESAT(Z, x_1, x_2)$ is a predicate indicating that Z is an assignment and the characters in positions $x_1 - x_2$ form a well formed quantifier-free subformula, which is satisfied by the Z assignment for a structure which is the whole formula.

Lemma 28. *$ATOMSAT$ and $QFREESAT$ can be written as formulas of $AA(3, 3)$ and $AA\Sigma(3, 4)$.*

Proof. For $ATOMSAT$ we write that $ATOM(x_1, x_2)$ and $ASS(Z)$ and that the assignment indeed satisfies the atomic formula. For example for an atomic formula $r_0(x)$ saying that x is the digit 0 we have

$$ATOM_{r_0}(x_1, x_2) \wedge ASS(Z) \wedge \exists y Z_0(x_1 + 2, y) \wedge r_0(y).$$

For an atomic formula $R^2(y_1, y_2)$ we have

$$ATOM_{R^2}(x_1, x_2) \wedge ASS(Z) \wedge \exists p_1, p_2, z_1, z_2(POS_1(x_1, p_1) \wedge POS_2(x_1, p_2) \wedge Z_0(p_1, z_1) \wedge Z_0(p_2, z_2) \wedge Z_3(x_1, z_1, z_2)).$$

For $QFREESAT$ we add a binary relation which gives the meaning function of well formed subformulas:

$$\exists R \forall y_1, y_2 [R(x_1, x_2) \wedge (R(y_1, y_2) \rightarrow (ATOMSAT(Z, y_1, y_2)$$
$$\vee (NOT(y_1, y_2) \wedge \neg R(y_1 + 1, y_2))$$
$$\vee (\exists y \; OR(y_1, y, y_2) \wedge (R(y_1 + 1, y - 1) \vee R(y + 1, y_2 - 1)))$$
$$\vee (\exists y \; AND(y_1, y, y_2) \wedge (R(y_1 + 1, y - 1) \wedge R(y + 1, y_2 - 1)))]. \qquad \square$$

Definition 29. (i) $\exists BLOCK(x_1, x_2, x_3)(\forall BLOCK(x_1, x_2, x_3))$ is a predicate indicating that the characters in positions $x_1 - x_2$ form an existential (universal) quantification of some variables and the characters in positions $x_2 + 1 - x_3$ form a well formed subformula.

(ii) $MODIF(Z^1, Z^2, x_1, x_2)$ is a predicate indicating that both Z^1 and Z^2 are assignments and that Z^1 is a modification of Z^2 for those variables which appear in the range $x_1 - x_2$ of the formula.

Lemma 30. *$\exists BLOCK$, $\forall BLOCK$ and $MODIF$ can be written in $AA(3, 3)$, $AA(3, 3)$ and $AA(2,4)$ respectively.*

Proof. $\exists BLOCK$ and $\forall BLOCK$ are context free. $MODIF(Z^1, Z^2, x_1, x_2)$ is a disjunction of formulas saying that if y_1 is an i-ary variable and is not the same as any i-ary variable which appears in the range $x_1 - x_2$ then its assignment in Z_i^1 is identical to its assignment in Z_i^2 $\square$

Definition 31. (i) $\exists BLOCKSAT_n(Z, x_1, x_2)$ is a predicate indicating that Z is a substitution and $x_1 - x_2$ form a formula with no more than n alternations of quantifiers, starting with $\exists$ which is satisfied by Z.

(ii) $\forall BLOCKSAT_n(Z, x_1, x_2)$ is a predicate indicating that Z is a substitution and $x_1 - x_2$ form a formula with no more than n alternations of quantifiers, starting with $\forall$ which is satisfied by Z.

Lemma 32. *$\exists BLOCKSAT_n(Z, x_1, x_2)$ and $\forall BLOCKSAT_n(Z, x_1, x_2)$ can be written in $AA(c, n + 4)$ where $c = \max(3, k + 1)$.*

Proof (*by induction on n*). For $n = 0$ $BLOCKSAT$ is just $QFREESAT$. $\exists BLOCKSAT_n$ can be written as

$$\exists Z_1, x \; (\exists BLOCK(x_1, x, x_2)$$
$$\wedge (MODIF(Z_1, Z, x_1, x) \wedge \forall BLOCKSAT_{n-1}(Z_1, x + 1, x_2))).$$

$\forall BLOCKSAT_n$ can be written as

$$\forall Z_1 \exists x\ (\forall BLOCK(x_1, x, x_2)$$

$$\wedge (MODIF(Z_1, Z, x_1, x) \wedge \exists BLOCKSAT_{n-1}(Z_1, x+1, x_2))). \qquad \square$$

To conclude the proof of Theorem C we have to show that

(ii) $AUTOSAT(FA(n))$ is definable in $AA(3, n+4)$ and

(iii) $AUTOSAT(AA(k, n))$ is definable in $AA(k+1, n+4)$.

The required formulas are simply conjunctions of the $2n$ $BLOCK_n$ formulas above.

6. Conclusions

We have studied the effect of bounding both arity and quantifier alternations in second-order formulas on their expressive power on finite structures. We have shown that the resulting hierarchies are proper.

The method to show this consisted of considering formulas which, viewed as finite structures, satisfy themselves. As the well-known diagonalization argument applies, this gives rise to the class $AUTOSAT(F)$ for sets of formulas F, which is not definable in F, provided F is closed under boolean operations.

We have given tight upper bounds for the complexity of $AUTOSAT(F)$ for various F. In particular, for $F = FOL$, the set of first-order formulas, this problem is *PSpace*-complete.

We would like to conclude with an open problem:

Problem 33. (i) Find a natural problem which is definable in $SAA\Sigma(3, 1)$ but not in $SAA\Sigma(2, 1)$.

(ii) Find a natural problem which is definable in $AA\Sigma(n+d, 1)$ but not in $SAA\Sigma(n, 1)$ for $2 \leqslant n$ and d as small as possible, $1 \leqslant d$.

(iii) More generally, find a natural problem which is definable in $SAA\Sigma(n+d, k)$ but not in $SAA\Sigma(n, k)$ for fixed k.

A correction

The first author would like to add: In [3], Theorem 4.7 is incorrect, as its proof contradicts Proposition 3.5 and Theorem 2. The theorem can be corrected, if we limit the number of alternations of first-order quantifiers and TC^k operators or restrict the formulas to formulas in Immerman Normal Form.

Acknowledgements

We are indebted to R. Fagin, who provided us with Proposition 2, and to the participants of the 1994 Oberwolfach Meeting on Finite Model Theory for various useful comments. We are also indebted to B. Courcelle for his comments concerning Proposition 15, and to A. Litman for valuable discussions.

A preliminary version of this paper has appeared as [12].

References

[1] M. Ajtai and R. Fagin, Reachability is harder for directed than for undirected finite graphs, J. Symbolic Logic 55 1 (1990) 113–150.

[2] J.R. Büchi, Weak second-order arithmetic and finite automata. Z. Math. Logik Grundlag. Math. 6 (1960) 66–92.

[3] A. Calò and J.A. Makowsky, The Ehrenfeucht–Fraïssé games for transitive closure, in: A. Nerode and M. Taitsin, eds., Logical Foundation of Computer Science – Tver'92, Lecture Notes in Computer Science, Vol. 620 (Springer, Berlin, 1992) 57–68.

[4] B. Courcelle, The monadic second-order logic of graphs I: recognizable sets of finite graphs, Inform. and Comput. 85 (1990) 12–75.

[5] M. de Rougemont, Second-order and inductive definability on finite structures, Z. Math. Logik Grundlag. Math. 33 (1987) 47–63.

[6] J. Engelfriet, A regular characterization of graph languages definable in monadic second-order logic, Theoret. Comput. Sci. 88 (1991) 139–150.

[7] M. Grohe, Bounded-arity hierarchies in fixed point logics, Preprint, 1993.

[8] J.E. Hopcroft and J.D. Ullman, Introduction to Automata Theory, Languages and Computation. Addison-Wesley Series in Computer Science (Addison-Wesley, Reading, MA, 1980).

[9] D.S. Johnson, A catalog of complexity classes, in: J. van Leeuwen, ed., Handbook of Theoretical Computer Science, Vol. 1, Ch. 2 (Elsevier, Amsterdam, 1990).

[10] J.F. Lynch, Complexity classes and theories of finite models, Math. Systems Theory 15 (1982) 127–144.

[11] J.A. Makowsky and Y.B. Pnueli. Computable quantifiers and logics over finite structures, in: M. Krynicki, M. Mostowsky and L. Sczerba, eds., Logics, Models, and Computation, Vol. 1 (Kluwer Academic Publishers, Dordrecht, 1995) 313–357.

[12] J.A. Makowsky and Y.B. Pnueli, Arity vs. alternation in second-order definability, in: LFCS'94, Lecture Notes in Computer Science, Vol. 813 (Springer, Berlin 1994) 240–252.

[13] W. Thomas, Classifying regular events in symbolic logic, J. Comput. System Sci. 25 (1982) 360–376.

[14] G. Turán, On the definability of properties of finite graphs, Discrete Math. 49 (1984) 291–302.