Theory Comput. Systems **30**, 423–428 (1997)

Theory of Computing Systems

Complexity of Presburger Arithmetic with Fixed Quantifier Dimension

U. Schöning

Abt. Theoretische Informatik, Universität Ulm,
D-89069 Ulm, Germany

Abstract. It is shown that the decision problem for formulas in Presburger arithmetic with quantifier prefix $[\exists_1 \forall_2 \cdots \exists_m \forall^3]$ (for m odd) and $[\exists_1 \forall_2 \cdots \forall_m \exists^3]$ (for m even) is complete for the class Σ_m^P of the polynomial-time hierarchy. Furthermore, the prefix type $[\exists\forall\exists\exists]$ is complete for Σ_2^P, and the prefix type $[\exists\forall]$ is complete for NP. This improves results (and solves a problem left open) by Grädel [7].

1. Introduction

We assume familiarity with the standard classes in complexity theory, like P and NP, the classes Σ_m^P of polynomial-time hierarchy, and the notion of polynomial-time reduction (see [6] and [9]).

Let 3-CNF (resp. 3-DNF) denote the set of propositional formulas in conjunctive (resp. disjunctive) normal form where each clause consists of three literals. A literal is a Boolean variable or a negated Boolean variable.

The decision problem B_m consists of Boolean formulas of the form $F(X^1, \ldots, X^m)$ where each X^i is a separate sequence of Boolean variables, $X^i = (x_1^i, \ldots, x_{n_i}^i)$, such that

$$\exists X^1 \forall X^2 \cdots Q_m X^m \; F(X^1, X^2, \ldots, X^m)$$

is true. Notice that $X_1, \ldots, X_m$ are *all* the variables occurring in F. It is known that, for each $m \geq 1$, B_m is complete for Σ_m^P (see [12] and [13]).

Presburger arithmetic is the first-order theory of the natural numbers with addition. Let PA denote the set of formulas which are true in this interpretation. It is known that PA has double-exponential complexity on alternating Turing machines (see [4], [3], and [1]) whereas the complexity of PA_m, the set of true Presburger formulas with m quantifier

alternations, is roughly one exponential step lower than the general case (see [10], [5], and [7]).

Formulas with fixed dimension are obtained by fixing the quantifier prefix (and therefore also the number of alternations) to a certain type which we denote by $[Q_1 Q_2 \cdots Q_m]$, $Q_i \in \{\exists, \forall\}$. Grädel has obtained in his dissertation [7] several Σ_m^P-completeness results for decision problems of the form $[Q_1 Q_2 \cdots Q_k] \cap \mathrm{PA}, k > m$. More precisely, he shows that

$[\exists_1 \forall_2 \cdots \exists_m \exists^2 \forall^3] \cap \mathrm{PA}$ is complete for Σ_m^P (if m is odd),

$[\exists_1 \forall_2 \cdots \forall_m \forall^2 \exists^3] \cap \mathrm{PA}$ is complete for Σ_m^P (if m is even).

For the case $m = 1$ he can obtain a stronger result:

$[\exists\forall\forall] \cap \mathrm{PA}$ is complete for NP.

By swapping universal and existential quantifiers a dual Π_m^P-completeness result can, of course, be obtained. Furthermore, the same complexity status as listed above all have such prefix classes which extend the above ones by finitely many quantifiers and do not increase the number of alternations. Therefore, the complexity status of all but finitely many prefix types is resolved in terms of a completeness result in the polynomial-time hierarchy. Some prefix types remain open, especially Grädel poses the open problem what the status of $[\exists\forall] \cap \mathrm{PA}$ is.

In this paper we strengthen the above completeness results by including more prefix types, and we resolve thereby the complexity status of $[\exists\forall] \cap \mathrm{PA}$; it is NP-complete.

2. Main Result

Theorem. *For each $m \geq 1$, the language $[\exists_1 \forall_2 \cdots \exists_m \forall^3] \cap PA$ (for m odd) and $[\exists_1 \forall_2 \cdots \forall_m \exists^3] \cap PA$ (for m even) is Σ_m^P-complete.*

Proof. Membership in Σ_m^P is shown in [7] (relying on results in [8], [11], and [10]).

For the following we assume that m is odd. In this case the problem $B_m \cap 3\text{-CNF}$ is complete for Σ_m^P (see [12] and [13]). (For the case of m even, we need to consider the problem $B_m \cap 3\text{-DNF}$ instead. The proof in this case is virtually the same.)

Let $F = F(X^1, X^2, \ldots, X^m)$ be a formula in 3-CNF where the X^i are sequences of Boolean variables, $X^i = (x_1^i, \ldots, x_k^i)$. We assume without loss of generality that each variable sequence X^i consists of the same number of variables, namely k.

Let

$$p_1, p_2, \ldots, p_k$$

be the sequence of the first k primes. It is important to notice that this sequence can be constructed in polynomial-time, relative to the size of F.

For the intended reduction, we want to map the formula

$$\exists X^1 \forall X^2 \cdots \exists X^m \; F(X^1, X^2, \ldots, X^m)$$

to a formula in Presburger arithmetic of the following form

$$\exists z_1 \forall z_2 \cdots \exists z_m \; G(z_1, z_2, \ldots, z_m),$$

where the z_i are variables that represent natural numbers (and encode the assignments X^i) and G is a Presburger formula intended to check whether these assignments make F true. A Boolean assignment $(x_1, \ldots, x_k) \in \{0, 1\}^k$ will be represented by a number z that satisfies the set of modular equations

$$\begin{aligned} z &\equiv x_1 \pmod{p_1}, \\ z &\equiv x_2 \pmod{p_2}, \\ &\vdots \\ z &\equiv x_k \pmod{p_k}. \end{aligned}$$

The existence of such a $z < \prod_{j=1}^{k} p_j$ is guaranteed by the Chinese remainder theorem.

We need to construct a Presburger formula $A(z)$ that evaluates to true if and only if the number z correctly represents a Boolean assignment, in the sense above. We need to express that for $j = 1, \ldots, k$ it holds that $(z \bmod p_j) \in \{0, 1\}$. Therefore $A(z)$ has the following tentative form:

$$\bigwedge_{j=1}^{k} [(z \bmod p_j) \in \{0, 1\}].$$

Equivalently,

$$\bigwedge_{j=1}^{k} \bigwedge_{l=2}^{p_j - 1} [z \not\equiv l \pmod{p_j}].$$

The expression in brackets can be rewritten as a formula in Presburger arithmetic:

$$\forall u \, (p_j \cdot u + l \neq z),$$

where the notation $p_j \cdot u$ is an abbreviation for $u + u + \cdots + u$ (p_j times). The universal quantifier can be pulled in front, so that the formula for $A(z)$ gets the final form

$$\forall u \left[\bigwedge_{j=1}^{k} \bigwedge_{l=2}^{p_j - 1} (p_j \cdot u + l \neq z) \right].$$

The intended formula

$$\exists z_1 \forall z_2 \cdots Q_m z_m \; G(z_1, z_2, \ldots, z_m)$$

is indeed equivalent to the following form:

$$\begin{array}{llll} \exists z_1 & (A(z_1) & \wedge & \\ & \forall z_2 \quad (A(z_2) & \rightarrow & \\ & \quad \exists z_3 \quad (A(z_3) & \wedge & \\ & & \ddots & \\ & & & H(z_1, z_2, \ldots, z_m) \ldots)). \end{array}$$

We have seen that $A(z_i)$ can be expressed by one universal quantifier. This enables us in this case to melt together quantifiers of the same type into one quantifier. (Melting quantifiers of the same type means that we are able to use just one quantifier of the same type where the quantified variable plays the role of the different original variables). In particular, the universal quantifier in $A(z_1)$ can be melted together with "$\forall z_2$" since they are connected conjunctively. Similarly, the existential quantifier that we need to express "$A(z_2) \rightarrow \cdots$" can be melted together with "$\exists z_3$" since they are connected by implication, and so on. Altogether, we get a quantifier prefix (before the beginning of the formula H) of the form $[\exists_1 \forall_2 \cdots \exists_m \forall]$.

The Presburger formula H is intended to express the fact that F is satisfied by the assignments $(X^1, \ldots, X^m)$. This formula consists of a conjunction of formulas,

$$H = \bigwedge_{i=1}^{n} C_i,$$

where C_i expresses in Presburger arithmetic that the ith clause in F is satisfied. Let the ith clause in F be $(x_{i_1}^{j_1})^{a_1} \vee (x_{i_2}^{j_2})^{a_2} \vee (x_{i_3}^{j_3})^{a_3}$ where $a_\mu \in \{0, 1\}$, $i_\mu \in \{1, \ldots, k\}$, $j_\mu \in \{1, \ldots, m\}$ for $\mu = 1, 2, 3$. Here $(x)^0$ means x and $(x)^1$ means $\neg x$. Then C_i is defined (tentatively) as

$$\neg[(z_{j_1} \equiv a_1 \pmod{p_{i_1}}) \wedge (z_{j_2} \equiv a_2 \pmod{p_{i_2}}) \wedge (z_{j_3} \equiv a_3 \pmod{p_{i_3}})],$$

which is equivalent to the Presburger formula

$$\forall u \forall v \forall w \, [(p_{i_1} \cdot u + a_1 \neq z_{j_1}) \vee (p_{i_2} \cdot u + a_2 \neq z_{j_2}) \vee (p_{i_3} \cdot u + a_3 \neq z_{j_3})].$$

In the case that the clause contains two (or three) variables with the same superscript (i.e., the variables come from the same block of variables X^l, $l \in \{1, \ldots, m\}$), then the formula C_i can be further simplified so that only two universal quantifiers (resp. one quantifier) is sufficient.

We best explain this with an example. Let the ith clause in F be $x_3^1 \vee \neg x_1^2 \vee x_2^2$. As above, we can, tentatively, express C_i as

$$\neg[(z_1 \equiv 0 \pmod{p_3}) \wedge (z_2 \equiv 1 \pmod{p_1}) \wedge (z_2 \equiv 0 \pmod{p_2})].$$

There are subformulas which start with the same z_i. By the Chinese remainder theorem, there is a number $a < p_1 p_2$ (which can be efficiently computed, see p. 824 of [2]) such that the above formula is equivalent to

$$\neg[(z_1 \equiv 0 \pmod{p_1}) \wedge (z_2 \equiv a \pmod{p_1 p_2})].$$

This can be expressed in Presburger arithmetic as

$$\begin{aligned}&\neg[\exists u(p_1 \cdot u = z_1) \wedge \exists v(p_1 p_2 \cdot v + a = z_2)]\\ &\quad \equiv \forall u \forall v[(p_1 \cdot u \neq z_1) \vee (p_1 p_2 \cdot v + a \neq z_2)].\end{aligned}$$

Like in this specific example we can, in general, express C_i by a formula with one, two, or three universal quantifiers, depending on the number of Boolean variables from different block X^i's (that is, the number of different z_i's) in that clause.

Up to three of these universal quantifiers per clause can be moved in front of the whole conjunction (and melted together) such that H gets the following form:

$$\forall u \forall v \forall w \left[\bigwedge_{i=1}^{n} (\cdots)\right].$$

Now the whole resulting formula has the following structure:

$$\exists z_1 \forall z_2 \cdots \exists z_m (A(z_m) \wedge \forall u \forall v \forall w [\cdots]).$$

Again we can melt together two quantifiers, namely the universal quantifier in $A(z_m)$ and "$\forall u$." So the final form of the Presburger formula has the quantifier prefix $[\exists_1 \forall_2 \cdots \exists_m \forall^3]$. Finally, we remark that the reduction can be carried out in polynomial time. □

Inspecting the proof, by the fact that the number of different X^i's determines the last block of universal quantifiers, we get the following corollary:

Corollary. *The decision problem* $[\exists\forall\exists\exists] \cap PA$ *is* Σ_2^P*-complete. The decision problem* $[\exists\forall] \cap PA$ *is NP-complete.*

Remarks. The NP-completeness of $[\exists\forall] \cap$ PA could also be obtained by a direct reduction from some other NP-complete problem; for this, Grädel (personal communication) has proposed problem [AN2] from [6].

The following remark originates in a suggestion by one of the anonymous referees. In the classic proof that Presburger arithmetic is decidable it is often actually proved that

a stronger language is decidable: PA + CONG, which allows symbols for the set

$$\{x \mid x \equiv a \pmod{b}\}$$

for any a, b. Hence this extension is natural. Since our proof uses these symbols (before they are transformed into an ordinary Presburger formula using an additional universal quantifier) a result about the complexity of PA + CONG, with fixed quantifier dimension, can be easily obtained. Inspecting the above proof, we see that in PA + CONG the formulas $A(z_i)$ are quantifier-free. The same holds for the formulas C_i. Therefore, for each $m \geq 1$, $[\exists_1 \forall_2 \cdots Q_m] \cap$ PA + CONG is complete for Σ_m^P.

References

[1] L. Berman. The complexity of logical theories. *Theoretical Computer Science* 11 (1980), 71–77.

[2] T.H. Cormen, C.E. Leiserson, R.L. Rivest. *Introduction to Algorithms*. MIT Press, Cambridge, MA, 1990.

[3] J. Ferrante, C.W. Rackoff. *The Computational Complexity of Logical Theories*. Lecture Notes in Mathematics, Vol. 718, Springer-Verlag, Berlin, 1979.

[4] M.J. Fischer, M.O. Rabin. Super-exponential complexity of Presburger arithmetic. *SIAM–AMS Proceedings* 7 (1974), 27–41.

[5] M. Fürer. The complexity of Presburger arithmetic with bounded quantifier alternation depth. *Theoretical Computer Science* 18 (1982), 105–111.

[6] M.R. Garey, D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, San Francisco, CA, 1979.

[7] E. Grädel. The Complexity of Subclasses of Logical Theories. Dissertation, Universität Basel, 1987.

[8] H. Lenstra. Integer programming with a fixed number of variables. *Mathematics of Operations Research* 8 (1983), 538–548.

[9] C. Papadimitriou. *Computational Complexity*. Addison-Wesley, Reading, MA, 1994.

[10] C.R. Reddy, D.W. Loveland. Presburger arithmetic with bounded quantifier alternation. *Proceedings of the 10th ACM Symposium on Theory of Computing*, 1978, pp. 320–325.

[11] B. Scarpellini. Complexity of subcases of Presburger arithmetic. *Transactions of the AMS* 284 (1984), 203–218.

[12] L. Stockmeyer. The polynomial-time hierarchy. *Theoretical Computer Science* 3 (1977), 1–22.

[13] C. Wrathall. Complete sets and the polynomial-time hierarchy. *Theoretical Computer Science* 3 (1977), 23–33.

Received April 1995, *and in final form August* 1996.