On the complexity of quantified integer programming

Dmitry Chistikov

Centre for Discrete Mathematics and its Applications (DIMAP) &
Department of Computer Science,
University of Warwick, UK
d.chistikov@warwick.ac.uk

Abstract

Quantified integer programming is the the problem of deciding assertions of the form $\mathrm{Q}_k \boldsymbol{x}_k \dots \forall \boldsymbol{x}_2\ \exists \boldsymbol{x}_1 : A \cdot \boldsymbol{x} \geq \boldsymbol{c}$ where vectors of variables $\boldsymbol{x}_k, \dots, \boldsymbol{x}_1$ form the vector $\boldsymbol{x}$, all variables are interpreted over $\mathbb{N}$ (alternatively, over $\mathbb{Z}$), and A and $\boldsymbol{c}$ are a matrix and vector over $\mathbb{Z}$ of appropriate sizes. We show in this paper that quantified integer programming with alternation depth k is complete for the kth level of the polynomial hierarchy.

Keywords: integer programming, semi-linear sets, Presburger arithmetic, quantifier elimination.

Based on joint work with Christoph Haase (University of Oxford, UK), to appear in ICALP'17.

1 Introduction

The problem of integer programming is, given a system of linear inequalities $A \cdot \boldsymbol{x} \geq \boldsymbol{b}$, to decide whether there exists a solution for $\boldsymbol{x}$ in the non-negative integers. This problem has been studied for decades, and its 0–1 version (in which the components of $\boldsymbol{x}$ are constrained to be either 0 or 1) is one of Karp's seminal 21 NP-complete problems [5]. In this paper, we study quantified integer programming (QIP), an extension of integer programming where some of the variables can be quantified universally—so that its instances have the form

$$\mathrm{Q}_k \boldsymbol{x}_k\ \dots\ \forall \boldsymbol{x}_2.\ \exists \boldsymbol{x}_1 : A \cdot \boldsymbol{x} \geq \boldsymbol{c} \tag{1}$$

where $\mathrm{Q}_i \in \{\exists, \forall\}$ and $\boldsymbol{x}$ consists of all first-order variables appearing in the vectors $\boldsymbol{x}_i$.

Our main contribution is settling the complexity of QIP with k quantifier blocks (as above): we prove this problem complete for the kth level of the polynomial hierarchy, similarly to the quantified version of SAT.[1] We also show that QIP with an unbounded number of quantifier blocks is PSPACE-hard and decidable in $\mathsf{STA}(*, 2^{n^{O(1)}}, n) \subseteq \mathsf{EXPSPACE}$.[2]

Theorem 1. *Σ_k-IP is complete for Σ_k^{P} if k is odd, and Π_k-IP is complete for Π_k^{P} if k is even.*

Related work and discussion. While the decidability of QIP is immediate—it can be viewed as a syntactic fragment of Presburger arithmetic, the (decidable) first-order theory of the natural numbers with addition and order, in which matrix formulas are constrained to be conjunctions of linear inequalities—its computational complexity has been unknown. It is, of course, not difficult to see that QIP (and in fact Presburger arithmetic) is PSPACE-complete if the interpretation of every first-order variable x_i is restricted to an interval $[l_i, u_i]$ that is given as part of the input: $x_i \in [l_i, u_i]$; see, e.g., [7]. But if $x_i \in \mathbb{N}$, then the best known upper bounds seem to be $\mathsf{STA}(*, 2^{2^{n^{O(1)}}}, O(n)) \subseteq \text{2-EXPSPACE}$, the generic upper bound for deciding Presburger arithmetic [1], and the $(k-1)$th level of the weak EXP hierarchy for the fragment with k quantifier blocks [3]. The best known lower bound has been Π_2^{P}, established recently by the authors for Π_2-instances of QIP [2, Sec. 4.2].

It may be surprising, and certainly was to the authors, that the complexity of QIP, a natural decision problem, has not yet been established. The main reason is probably that standard quantifier-elimination and automata-based techniques—which are at the core of decision procedures for Presburger arithmetic—fail to yield tight upper bounds for QIP.

[1] As in the case of quantified CNF SAT, the innermost block of universal quantifiers, if present, is disregarded; e.g., the $\forall^*\exists^*\forall^*$ fragment is complete for Π_2^{P}. So we find fragments of QIP complete for $\Sigma_1^{\mathsf{P}} = \mathsf{NP}$, Π_2^{P}, Σ_3^{P}, ... , but not for $\mathsf{coNP} = \Pi_1^{\mathsf{P}}$, Σ_2^{P}, ...

[2] The complexity class $\mathsf{STA}(s(n), t(n), a(n))$ was introduced by Berman [1] and contains all decision problems that can be decided by an alternating Turing machine in time $t(n)$ using space at most $s(n)$ and alternating at most $a(n)$ times on every computation branch.

Our main results are, in short, obtained by means of a new quantifier elimination procedure on *hybrid linear sets*, which are semi-linear sets that represent sets of solutions to systems of linear inequalities. While *existential* projection ($L \mapsto \{x : \exists y.\ (x, y) \in L\}$) is a trivial operation on semi-linear sets (in generator representation), in this paper we define a dual operation, which we call *universal projection* ($L \mapsto \{x : \forall y.\ (x, y) \in L\}$), and show that its application enables us to eliminate blocks of universal quantifiers without resorting to double complementation ($\forall = \neg\exists\neg$; this would lead to a non-elementary blowup).

Concurrently with our work and building upon a theorem of Kannan [4], Nguyen and Pak [6] have shown that Presburger arithmetic with fixed number of variables *and* fixed Boolean structure of the matrix formula (and, by necessity, where the total number of occurrences of atomic predicates is fixed) can be solved in polynomial time.

Acknowledgements. This talk is based on joint work with Christoph Haase (University of Oxford, UK), to appear in ICALP'17. The work was done while Dmitry Chistikov was a research assistant at the University of Oxford, UK, supported by the ERC grant AVS-ISS (648701).

References

[1] Leonard Berman. The complexitiy of logical theories. *Theor. Comput. Sci.*, 11:71–77, 1980. doi:10.1016/0304-3975(80)90037-7.

[2] Dmitry Chistikov, Christoph Haase, and Simon Halfon. Context-free commutative grammars with integer counters and resets. *Theor. Comput. Sci.*, pages –, 2017. To appear. doi:10.1016/j.tcs.2016.06.017.

[3] Christoph Haase. Subclasses of Presburger arithmetic and the weak EXP hierarchy. In *Computer Science Logic and Logic in Computer Science, CSL-LICS*, pages 47:1–47:10. ACM, 2014. doi:10.1145/2603088.2603092.

[4] Ravi Kannan. Test sets for integer programs, $\forall\exists$ sentences. In *Polyhedral Combinatorics, Proceedings of a DIMACS Workshop, Morristown, New Jersey, USA, June 12-16, 1989*, pages 39–48, 1990.

[5] Richard M. Karp. Reducibility among combinatorial problems. In *Complexity of Computer Computations*, The IBM Research Symposia Series, pages 85–103. Plenum Press, New York, 1972.

[6] Danny Nguyen and Igor Pak. Complexity of short Presburger arithmetic. In *49th Annual ACM Symposium on the Theory of Computing, STOC*, 2017. To appear.

[7] K. Subramani. Tractable fragments of Presburger arithmetic. *Theory Comput. Syst.*, 38(5):647–668, 2005. doi:10.1007/s00224-004-1220-0.