The Journal of Symbolic Logic
Volume 80, Number 2, June 2015

PRESBURGER ARITHMETIC, RATIONAL GENERATING FUNCTIONS, AND QUASI-POLYNOMIALS

KEVIN WOODS

Abstract. Presburger arithmetic is the first-order theory of the natural numbers with addition (but no multiplication). We characterize sets that can be defined by a Presburger formula as exactly the sets whose characteristic functions can be represented by rational generating functions; a geometric characterization of such sets is also given. In addition, if $\mathbf{p} = (p_1, \ldots, p_n)$ are a subset of the free variables in a Presburger formula, we can define a counting function $g(\mathbf{p})$ to be the number of solutions to the formula, for a given $\mathbf{p}$. We show that every counting function obtained in this way may be represented as, equivalently, either a piecewise quasi-polynomial or a rational generating function. Finally, we translate known computational complexity results into this setting and discuss open directions.

§1. Introduction. A broad and interesting class of sets are those that can be defined over $\mathbb{N} = \{0, 1, 2, \ldots\}$ with first order logic and addition.

Definition 1.1. A *Presburger formula* is a first-order formula in the language of addition, evaluated over the natural numbers. We will denote a generic Presburger formula as $F(\mathbf{u})$, where $\mathbf{u}$ are the free variables (those not associated with a quantifier); we use bold notation like $\mathbf{u}$ to indicate vectors of variables.

We say that a set $S \subseteq \mathbb{N}^d$ is a *Presburger set* if there exists a Presburger formula $F(\mathbf{u})$ such that $S = \{\mathbf{u} \in \mathbb{N}^d : F(\mathbf{u})\}$.

Example 1.2. The Presburger formula

$$F(u) = \big(u > 1 \text{ and } \exists b \in \mathbb{N} : b + b + 1 = u\big)$$

defines the Presburger set $\{3, 5, 7, \ldots\}$. Since multiplication by an integer is the same as repeated addition, we can conceive of a Presburger formula as a Boolean combination of integral linear (in)equalities, appropriately quantified: $\exists b \ \big(u > 1$ and $2b + 1 = u\big)$.

Presburger proved [40] that the truth of a Presburger *sentence* (a formula with no free variables) is decidable. We would like to understand more clearly the *structure*

Received July 30, 2013.
Key words and phrases. Ehrhart polynomials, generating functions, Presburger arithmetic, quasi-polynomials, computational complexity.
Extended abstract published in ICALP 2013.

© 2015, Association for Symbolic Logic
0022-4812/15/8002-0003
DOI:10.1017/jsl.2015.4

https://doi.org/10.1017/jsl.2015.4 Published online by Cambridge University Press

of a given Presburger set. One way to attempt to do this is to encode the elements of the set into a generating function.

DEFINITION 1.3. Given a set $S \subseteq \mathbb{N}^d$, its associated *generating function* is

$$f(S;\mathbf{x}) = \sum_{\mathbf{s}\in S} \mathbf{x}^{\mathbf{s}} = \sum_{(s_1,\ldots,s_d)\in S} x_1^{s_1} x_2^{s_2} \cdots x_d^{s_d}.$$

For example, if S is the set defined by Example 1.2, then

$$f(S;x) = x^3 + x^5 + x^7 + \cdots = \frac{x^3}{1-x^2}.$$

We see that, in this instance, the generating function has a nice form; this is not a coincidence.

DEFINITION 1.4. A *rational generating function* is a function that can be written in the form

$$\frac{q(\mathbf{x})}{(1-\mathbf{x}^{\mathbf{b}_1})\cdots(1-\mathbf{x}^{\mathbf{b}_k})},$$

where $q(\mathbf{x})$ is a polynomial in $\mathbb{Q}[\mathbf{x}]$ and $\mathbf{b}_i \in \mathbb{N}^d \setminus \{\mathbf{0}\}$.

We will prove that $S \subseteq \mathbb{N}^d$ is a Presburger set if and only if $f(S;\mathbf{x})$ is a rational generating function. These are Properties 1 and 3 in the following theorem:

THEOREM 1.5. *Given a set $S \subseteq \mathbb{N}^d$, the following are equivalent*:

1. *S is a Presburger set*,
2. *S is a finite union of sets of the form $P \cap (\lambda + \Lambda)$, where P is a polyhedron, $\lambda \in \mathbb{Z}^d$, and $\Lambda \subseteq \mathbb{Z}^d$ is a lattice.*
3. *$f(S;\mathbf{x})$ is a rational generating function*.

Property 2 gives a nice geometric characterization of Presburger sets; the set in Example 1.2 can be written as $[3,\infty) \cap (1+2\mathbb{Z})$.

We are particularly interested in generating functions because of their powerful flexibility: we can use algebraic manipulations to answer questions about the set. For example, $f(S;1,1,\ldots,1)$ is exactly the cardinality of S (if finite). More generally, we may want to count solutions to a Presburger formula as a function of several parameter variables:

DEFINITION 1.6. The *Presburger counting function* for a Presburger formula $F(\mathbf{c},\mathbf{p})$ is

$$g_F(\mathbf{p}) = \#\{\mathbf{c} \in \mathbb{N}^d : F(\mathbf{c},\mathbf{p})\}.$$

Note that $\mathbf{c}$ (the *counted* variables) and $\mathbf{p}$ (the *parameter* variables) are free variables. We will restrict ourselves to counting functions such that $g_F(\mathbf{p})$ is finite for all $\mathbf{p} \in \mathbb{N}^n$. One could instead either include ∞ in the codomain of g_F or restrict the domain of g_F to where $g_F(\mathbf{p})$ is finite (the domain would itself be a Presburger set).

A classic example is to take $F(\mathbf{c}, p)$ to be the conjunction of linear inequalities of the form $a_1c_1 + \cdots + a_dc_d \leq a_0p$, where $a_i \in \mathbb{Z}$. Then $g_F(p)$ counts the number of integer points in the p^{th} dilate of a polyhedron.

https://doi.org/10.1017/jsl.2015.4 Published online by Cambridge University Press

EXAMPLE 1.7. If $F(c_1, c_2, p)$ is $2c_1 + 2c_2 \leq p$, then the set of solutions $(c_1, c_2) \in \mathbb{N}^2$ lies in the triangle with vertices $(0, 0)$, $(0, p/2)$, $(p/2, 0)$, and

$$g_F(p) = \frac{1}{2}\left(\left\lfloor \frac{p}{2} \right\rfloor + 1\right)\left(\left\lfloor \frac{p}{2} \right\rfloor + 2\right)$$
$$= \begin{cases} \frac{1}{8}p^2 + \frac{3}{4}p + 1 & \text{if } p \text{ is even,} \\ \frac{1}{8}p^2 + \frac{1}{2}p + \frac{3}{8} & \text{if } p \text{ is odd.} \end{cases}$$

The nice form of this function is also not a coincidence. For this particular type of Presburger formula (dilates of a polyhedron), Ehrhart proved [21] that the counting functions are *quasi-polynomials*:

DEFINITION 1.8. A *quasi-polynomial* (over $\mathbb{Q}$) is a function $g : \mathbb{N}^n \to \mathbb{Q}$ such that there exists an n-dimensional lattice $\Lambda \subseteq \mathbb{Z}^n$ together with polynomials $q_{\bar{\lambda}}(\mathbf{p}) \in \mathbb{Q}[\mathbf{p}]$, one for each $\bar{\lambda} \in \mathbb{Z}^n/\Lambda$, such that

$$g(\mathbf{p}) = q_{\bar{\lambda}}(\mathbf{p}), \text{ for } \mathbf{p} \in \bar{\lambda}.$$

In Example 1.7, we can take the lattice $\Lambda = 2\mathbb{Z}$ and each coset (the evens and the odds) has its associated polynomial. We need something slightly more general to account for all Presburger counting functions:

DEFINITION 1.9. A *piecewise quasi-polynomial* is a function $g : \mathbb{N}^n \to \mathbb{Q}$ such that there exists a finite partition $\bigcup_i (P_i \cap \mathbb{N}^n)$ of $\mathbb{N}^n$ with P_i polyhedra (which may not all be full-dimensional) and there exist quasi-polynomials g_i such that

$$g(\mathbf{p}) = g_i(\mathbf{p}) \text{ for } \mathbf{p} \in P_i \cap \mathbb{N}^n.$$

One last thing that is not a coincidence: For the triangle in Example 1.7, we can compute

$$\sum_{p \in \mathbb{N}} g_F(p)x^p = 1 + x + 3x^2 + 3x^3 + 6x^4 + \cdots$$
$$= \frac{1}{(1-x)(1-x^2)^2},$$

a rational generating function! The following theorem says that these ideas are – almost – equivalent.

THEOREM 1.10. *Given a function* $g : \mathbb{N}^n \to \mathbb{Q}$ *and the following three possible properties*:

A. *g is a Presburger counting function*,
B. *g is a piecewise quasi-polynomial, and*
C. $\sum_{\mathbf{p} \in \mathbb{N}^n} g(\mathbf{p})\mathbf{x}^{\mathbf{p}}$ *is a rational generating function*,

we have the implications

$$A \Rightarrow B \Leftrightarrow C.$$

REMARK 1.11. Proving Theorem 1.10 will give us much of Theorem 1.5, using the following idea. A set $S \subseteq \mathbb{Z}^d$ corresponds exactly to its characteristic function

$$\chi_S(\mathbf{u}) = \begin{cases} 1 & \text{if } \mathbf{u} \in S, \\ 0 & \text{if } \mathbf{u} \notin S. \end{cases}$$

https://doi.org/10.1017/jsl.2015.4 Published online by Cambridge University Press

If S is a Presburger set defined by $F(\mathbf{u})$, then

$$\chi_S(\mathbf{u}) = \#\{c \in \mathbb{N} : \ F(\mathbf{u}) \text{ and } c = 0\}$$

is a Presburger counting function.

In light of Theorem 1.5, we might wonder if there is a sense in which $B \Rightarrow A$. Of course we would have to restrict g, for example requiring that its range be in $\mathbb{N}$ (Theorem 1.5 essentially restricts the range of g to $\{0, 1\}$, as it must be a characteristic function). The implication still does not hold, however. For example, suppose the polynomial

$$g(s, t) = (t - s^2)^2$$

were a Presburger counting function given by a Presburger formula $F(\mathbf{c}, s, t)$, that is,

$$g(s, t) = \#\{\mathbf{c} \in \mathbb{N}^d : \ F(\mathbf{c}, s, t)\}.$$

Then the set

$$\begin{aligned}\{(s, t) \in \mathbb{N}^2 : \ \nexists \mathbf{c}\, F(\mathbf{c}, s, t)\} &= \{(s, t) \in \mathbb{N}^2 : \ g(s, t) = 0\} \\ &= \{(s, s^2) : \ s \in \mathbb{N}\}\end{aligned}$$

would be a Presburger set. This is not the case, however, as it does not satisfy Property 2 in Theorem 1.5. If the parameter is univariate, however, the following proposition shows that we do have the implication $B \Rightarrow A$.

PROPOSITION 1.12. *Given a function $g : \mathbb{N} \to \mathbb{Q}$, if g is a piecewise quasi-polynomial whose range is in $\mathbb{N}$, then g is a Presburger counting function.*

In Section 4, we prove Theorem 1.5, Theorem 1.10, and Proposition 1.12. In Section 2, we survey related work. In Section 3, we present the primary tools we need for the proofs. In Section 5, we turn to computational questions; this section surveys known results, but restates them in terms of Presburger arithmetic.

§2. Related work. Presburger arithmetic is a classical first order theory of logic, proven decidable by Presburger [40]. Bounds on the complexity of decision algorithms have been examined in the general case [8, 17, 22, 23, 38], as a function of the number of quantifier alternations [25, 28, 31, 43, 46], and for fixed (small) number of quantifier alternations [29, 45].

A finite automata approach to Presburger arithmetic was pioneered in [12, 15], and continues to be an active area of research (see, for example, [10, 16, 34, 55]). This approach is quite different from the present paper's, but it can attack similar questions: for example, see [39] for results on counting solutions to Presburger formulas (nonparametrically).

The importance of understanding Presburger Arithmetic is highlighted by the fact that many problems in computer science and mathematics can be phrased in this language: for example, integer programming [35, 48], geometry of numbers [13, 33], Gröbner bases and algebraic integer programming [51, 53], neighborhood complexes and test sets [44, 52], the Frobenius problem [42], Ehrhart theory [7, 21], monomial ideals [37], and toric varieties [24]. Several of the above references analyze the computational complexity of their specific problem. In most of the above references, the connection to Presburger arithmetic is only implicit.

https://doi.org/10.1017/jsl.2015.4 Published online by Cambridge University Press

The algorithmic complexity of specific rational generating function problems has been addressed in, for example, [1, 5, 9, 20, 30, 32]. Several of these results are summarized in Section 5 of this current paper.

Connections between subclasses of Presburger arithmetic and generating functions are made explicit in [3–5]. Connections between rational generating functions and quasi-polynomials have been made in [21, 49, 50], and the algorithmic complexity of their relationship was examined in [54]. Counting solutions to Presburger formulas have been examined in [41], though the exact scope of the results is not made explicit, and rational generating functions are not used. Similar counting algorithms appear in [14], and [18] proves that the counting functions for a special class of Presburger formulas (those whose parameters $\mathbf{p}$ only appear in terms $c_i \leq p_i$) are piecewise quasi-polynomials. This current paper is the first to state and prove a general connection between Presburger arithmetic, quasi-polynomials, and rational generating functions.

Theorem 1.5 was originally proved in the author's thesis [56]; in this paper, it is put into context as a consequence of the more general Theorem 1.10. A simpler geometric characterization of Presburger sets (equivalent to Property 2 of Theorem 1.5) was given in [26]: they are the *semilinear* sets, those sets that can be written as a finite union of sets of the form $S = \{\mathbf{a}_0 + \sum_{i=1}^{k} n_i \mathbf{a}_i : n_i \in \mathbb{N}\}$, where $\mathbf{a}_i \in \mathbb{N}^d$. Furthermore, if one takes these S to be disjoint and requires the $\mathbf{a}_1, \ldots, \mathbf{a}_k$ to be linearly independent, for each S (as [26] implicitly prove can be done, made explicit in [18] as *semisimple* sets), then each S can be encoded with the rational generating function

$$\frac{\mathbf{x}^{\mathbf{a}_0}}{(1-\mathbf{x}_1^{\mathbf{a}_1})\cdots(1-\mathbf{x}_k^{\mathbf{a}_k})}$$

and we obtain a slightly different version of $2 \Rightarrow 3$ in Theorem 1.5. There seems to be no previous result analogous to $3 \Rightarrow 2$.

An extended abstract of this current paper appeared in ICALP 2013.

§3. **Primary background theorems.** Here we detail several tools we will use. When these tools have algorithmic (polynomial-time) versions, we mention them too, which will help our discussion in Section 5.

The first tool we need is a way to simplify Presburger formulas. As originally proved [40] by Presburger (see [38] for a nice exposition), we can completely eliminate the quantifiers if we are allowed to also use modular arithmetic.

DEFINITION 3.1. An *extended Presburger formula* is a Boolean formula with variables in $\mathbb{N}$ expressible in the elementary language of Presburger Arithmetic extended by the mod k operations, for constants $k > 1$.

THEOREM 3.2. *Given a formula $F(\mathbf{u})$ in extended Presburger arithmetic (and hence given any formula in Presburger arithmetic), there exists an equivalent* quantifier-free *formula $G(\mathbf{u})$ such that*

$$\{\mathbf{u} \in \mathbb{N}^d : F(\mathbf{u})\} = \{\mathbf{u} \in \mathbb{N}^d : G(\mathbf{u})\}.$$

For instance, the set from Example 1.2 can be written as $(u > 1 \text{ and } u \bmod 2 = 1)$.

https://doi.org/10.1017/jsl.2015.4 Published online by Cambridge University Press

Next, we give two theorems that tie in generating functions. The first gives us a way to convert from a specific type of Presburger set to a generating function.

THEOREM 3.3. *Given a point $\lambda \in \mathbb{Z}^d$, a lattice $\Lambda \subseteq \mathbb{Z}^d$, and a rational polyhedron $P \subseteq \mathbb{R}^d_{\geq 0}$, $f\left(P \cap (\lambda + \Lambda); \mathbf{x}\right)$ (as given in Definition 1.3) is a rational generating function. In addition, for fixed d, this rational generating function can be found in polynomial time.*

The first step to prove this is to use Brion's Theorem [11], which says that the generating function can be decomposed into functions of the form $f\left(K \cap (\lambda + \Lambda); \mathbf{x}\right)$, where K is a cone. Then, one can notice that integer points in cones have a natural structure that can be encoded as geometric series.

EXAMPLE 3.4. Let $K \subseteq \mathbb{R}^2$ be the cone with vertex at the origin and extreme rays $\mathbf{u} = (1, 0)$ and $\mathbf{v} = (1, 2)$. Using the fact that the lattice $(u\mathbb{Z} + v\mathbb{Z})$ has index 2 in $\mathbb{Z}^2$, with coset representatives $(0, 0)$ and $(1, 1)$, every integer point in K can be written as either $(0, 0) + \lambda_1\mathbf{u} + \lambda_2\mathbf{v}$ or $(1, 1) + \lambda_1\mathbf{u} + \lambda_2\mathbf{v}$, where $\lambda_1, \lambda_2 \in \mathbb{N}$. Therefore

$$f(K \cap \mathbb{Z}^2; \mathbf{x}) = (\mathbf{x}^{(0,0)} + \mathbf{x}^{(1,1)})(1 + \mathbf{x}^{\mathbf{u}} + \mathbf{x}^{2\mathbf{u}} + \cdots)(1 + \mathbf{x}^{\mathbf{v}} + \mathbf{x}^{2\mathbf{v}} + \cdots)$$
$$= \frac{\mathbf{x}^{(0,0)} + \mathbf{x}^{(1,1)}}{(1 - \mathbf{x}^{\mathbf{u}})(1 - \mathbf{x}^{\mathbf{v}})}.$$

See [2, Chapter VIII], for example, for more details.

The complexity version of Theorem 3.3 is due to Barvinok [1]. For polynomial-time algorithms yielding generating functions, the output will generally be represented as a *sum* of the basic rational functions given in Definition 1.4; getting a common denominator and simplifying into one fraction may take exponential time. Since the dimension is fixed, the input polyhedra may equivalently be given with either a vertex or hyperplane description.

Next, we would like to be able to perform substitutions on the variables in a rational generating function and still retain a rational generating function; particularly, we would like to substitute in 1's for several of the variables.

THEOREM 3.5. *Given a rational generating function $f(\mathbf{x})$, then*

$$g(\mathbf{z}) = f(\mathbf{z}^{\mathbf{l}_1}, \mathbf{z}^{\mathbf{l}_2}, \ldots, \mathbf{z}^{\mathbf{l}_d}),$$

with $\mathbf{l}_i \in \mathbb{N}^k$, is also a rational generating function, assuming the substituted values do not lie entirely in the poles of f. In particular, substituting in $x_i = \mathbf{z}^{\mathbf{0}} = 1$ yields a rational function, if $x_i = 1$ is not entirely in the poles of f.

Furthermore, for a fixed bound on the number of binomials in the denominator of f, there is a polynomial time algorithm to compute g as a rational generating function.

The existence version of this theorem is immediate: if substituting in $x_i = \mathbf{z}^{\mathbf{l}_i}$ would make any of the binomials in the denominator of f zero (when f is written in the form from Definition 1.4), then that binomial must be a factor of the numerator (or else $x_i = \mathbf{z}^{\mathbf{l}_i}$ would lie entirely in the poles of f); therefore, substituting in $x_i = \mathbf{z}^{l_i}$ yields a new rational generating function. The complexity version is more difficult and is due to Barvinok and Woods [5]. The problem is that we may have the rational generating function input as a sum of rational functions. Unfortunately,

https://doi.org/10.1017/jsl.2015.4 Published online by Cambridge University Press

the substituted values may be in the poles of some of the terms of the sum without being in the poles of the entire generating function; careful limits must be taken.

EXAMPLE 3.6. Suppose we would like to substitute $x = 1$ into

$$f(x) = \frac{1}{1-x} - \frac{x^{1000}}{1-x}.$$

Combining and simplifying to $f(x) = 1 + x + \cdots + x^{999}$ is horribly inefficient. Since $x = 1$ is a pole of both fractions, we must be careful: Consider the respective Laurent series expansions of the terms:

$$\frac{1}{1-x} = c_{-1}(x-1)^{-1} + c_0(x-1)^0 + c_1(x-1)^1 + \cdots,$$
$$-\frac{x^{1000}}{1-x} = d_{-1}(x-1)^{-1} + d_0(x-1)^0 + d_1(x-1)^1 + \cdots.$$

We know that 1 is not a pole of $f(x)$, so $c_{-1} + d_{-1} = 0$. Substituting in $x = 1$ into the sum, we will get simply $c_0 + d_0$. Carefully calculating c_0 and d_0 will give us $f(1) = 1000$.

Finally, we need a connection between Presburger formulas and quasi-polynomials. This is given by Sturmfels [50]:

DEFINITION 3.7. Given $\mathbf{a}_1, \ldots, \mathbf{a}_d \in \mathbb{N}^n$, the *vector partition function* $g : \mathbb{N}^n \to \mathbb{N}$ is defined by

$$g(\mathbf{p}) = \#\{(\lambda_1, \ldots, \lambda_d) \in \mathbb{N}^d : \mathbf{p} = \lambda_1\mathbf{a}_1 + \cdots + \lambda_d\mathbf{a}_d\},$$

that is, the number of ways to partition the vector $\mathbf{p}$ into parts taken from $\{\mathbf{a}_i\}$.

THEOREM 3.8. *Any vector partition function is a piecewise quasi-polynomial.*

See [6] for a self-contained explanation utilizing the partial fraction expansion of the generating function

$$\sum_{\mathbf{p}\in\mathbb{N}^n} g(\mathbf{p})\mathbf{x}^{\mathbf{p}} = \frac{1}{(1-\mathbf{x}^{\mathbf{a}_1})\cdots(1-\mathbf{x}^{\mathbf{a}_d})};$$

this equality can be obtained by rewriting the rational function as a product of infinite geometric series:

$$(1 + \mathbf{x}^{\mathbf{a}_1} + \mathbf{x}^{2\mathbf{a}_1} + \cdots)\cdots(1 + \mathbf{x}^{\mathbf{a}_d} + \mathbf{x}^{2\mathbf{a}_d} + \cdots).$$

For example, if $a_1 = 1$ and $a_2 = a_3 = 2$, then the vector partition function is encoded by the generating function

$$\frac{1}{(1-x)(1-x^2)^2}.$$

We saw previously that this generating function corresponds to the quasi-polynomial in Example 1.7.

A complexity version of Theorem 3.8 is discussed in Section 5.2; an appropriate representation for piecewise quasi-polynomials must first be decided on.

https://doi.org/10.1017/jsl.2015.4 Published online by Cambridge University Press

§4. Proofs.

4.1. Proof of Theorem 1.10. A ⇒ C.

Given a Presburger counting function, $g(\mathbf{p}) = \#\{\mathbf{c} \in \mathbb{N}^d : F(\mathbf{c}, \mathbf{p})\}$, we first apply Presburger Elimination (Theorem 3.2) to F to obtain a quantifier free formula, $G(\mathbf{c}, \mathbf{p})$, in extended Presburger arithmetic such that $g(\mathbf{p}) = \#\{\mathbf{c} \in \mathbb{N}^d : G(\mathbf{c}, \mathbf{p})\}$. Integers which satisfy a statement of the form

$$a_1 p_1 + \cdots + a_n p_n + a_{n+1} c_1 + \cdots + a_{n+d} c_d \equiv a_0 \mod m$$

are exactly sets $\lambda + \Lambda$, where $\lambda \in \mathbb{Z}^{n+d}$ and Λ is a lattice in $\mathbb{Z}^{n+d}$. Since $G(\mathbf{c}, \mathbf{p})$ is a Boolean combination of linear inequalities and these linear congruences, we may write the set, S, of points $(\mathbf{c}, \mathbf{p})$ which satisfy $G(\mathbf{c}, \mathbf{p})$ as a *disjoint* union

$$S = \bigcup_{i=1}^{k} P_i \cap (\lambda_i + \Lambda_i),$$

where, for $1 \leq i \leq k$, $P_i \subseteq \mathbb{R}^{n+d}_{\geq 0}$ is a polyhedron, Λ_i is a sublattice of $\mathbb{Z}^{n+d}$, and λ_i is in $\mathbb{Z}^{n+d}$. (To see this, convert the formula into disjunctive normal form; each conjunction will be of this form $P_i \cap (\lambda_i + \Lambda_i)$; these sets may overlap, but their overlap will also be of this form.)

Let $S_i = P_i \cap (\lambda_i + \Lambda_i)$. By Theorem 3.3, we know we can write $f(S_i; \mathbf{y}, \mathbf{x})$ as a rational generating function, and so

$$f(S; \mathbf{y}, \mathbf{x}) = \sum_i f(S_i; \mathbf{y}, \mathbf{x}) = \sum_{(\mathbf{c},\mathbf{p}):\ G(\mathbf{c},\mathbf{p})} \mathbf{y}^{\mathbf{c}} \mathbf{x}^{\mathbf{p}}$$

can be written as a rational generating function. Finally, we substitute $\mathbf{y} = (1, 1, \ldots, 1)$, using Theorem 3.5, to obtain the rational generating function

$$\sum_{\mathbf{p}} \#\{\mathbf{c} \in \mathbb{N}^d : G(\mathbf{c}, \mathbf{p})\} \mathbf{x}^{\mathbf{p}} = \sum_{\mathbf{p}} g(\mathbf{p}) \mathbf{x}^{\mathbf{p}}.$$

C ⇒ B.

It suffices to prove this for functions g such that $\sum_{\mathbf{p}} g(\mathbf{p}) \mathbf{x}^{\mathbf{p}}$ is a rational generating function of the form

$$\frac{\mathbf{x}^{\mathbf{q}}}{(1 - \mathbf{x}^{\mathbf{a}_1})(1 - \mathbf{x}^{\mathbf{a}_2}) \cdots (1 - \mathbf{x}^{\mathbf{a}_k})},$$

where $\mathbf{q} \in \mathbb{N}^n, \mathbf{a}_i \in \mathbb{N}^n \setminus \{0\}$, because the property of being a piecewise quasi-polynomial is preserved under linear combinations. Furthermore, we may take $\mathbf{q} = (0, 0, \ldots, 0)$, because multiplying by $\mathbf{x}^{\mathbf{q}}$ only shifts the domain of the function g. Expanding this rational generating function as a product of infinite geometric series,

$$\sum_{\mathbf{p}} g(\mathbf{p}) \mathbf{x}^{\mathbf{p}} = (1 + \mathbf{x}^{\mathbf{a}_1} + \mathbf{x}^{2\mathbf{a}_1} + \cdots) \cdots (1 + \mathbf{x}^{\mathbf{a}_k} + \mathbf{x}^{2\mathbf{a}_k} + \cdots),$$

and we see that

$$g(\mathbf{p}) = \#\{(\lambda_1, \ldots, \lambda_k) \in \mathbb{N}^k : \mathbf{p} = \lambda_1 \mathbf{a}_1 + \cdots + \lambda_k \mathbf{a}_k\}.$$

This is exactly a vector partition function, which Theorem 3.8 tells us is a piecewise quasi-polynomial.

https://doi.org/10.1017/jsl.2015.4 Published online by Cambridge University Press

B $\Rightarrow$ **C**.

Any piecewise quasi-polynomial can be written as a linear combination of functions of the form

$$g(\mathbf{p}) = \begin{cases} \mathbf{p}^{\mathbf{a}} & \text{if } \mathbf{p} \in P \cap (\lambda + \Lambda), \\ 0 & \text{otherwise,} \end{cases}$$

where $\mathbf{a} \in \mathbb{N}^n$, $P \subseteq \mathbb{R}^n_{\geq 0}$ is a polyhedron, $\lambda \in \mathbb{Z}^n$, and Λ is a sublattice of $\mathbb{Z}^n$. Since linear combinations of rational generating functions are rational generating functions, it suffices to prove it for such a g. Let c_{ij}, for $1 \leq i \leq n$ and $1 \leq j \leq a_i$, be variables, and define the polyhedron

$$Q = \{(\mathbf{p}, \mathbf{c}) \in \mathbb{N}^{n+a_1+\cdots+a_n} : \\ \mathbf{p} \in P \text{ and } 1 \leq c_{ij} \leq p_i \text{ for all } c_{ij}\}.$$

This Q is defined so that $\#\{\mathbf{c} : (\mathbf{p}, \mathbf{c}) \in Q\}$ is $p_1^{a_1} \cdots p_n^{a_n} = \mathbf{p}^{\mathbf{a}}$ for $\mathbf{p} \in P$ (and 0 otherwise). Using Theorem 3.3, we can find the generating function for the set $Q \cap (\lambda + \Lambda)$ as a rational generating function. Substituting $\mathbf{c} = (1, 1, \ldots, 1)$, using Theorem 3.5, gives us $\sum_{\mathbf{p}} g(\mathbf{p})\mathbf{x}^{\mathbf{p}}$ as a rational generating function.

4.2. Proof of Theorem 1.5. Given a set $S \subseteq \mathbb{Z}^d$, define the characteristic function, $\chi_S : \mathbb{N}^d \to \{0, 1\}$, as in Remark 1.11. Define a new property:

$2'$. χ_S is a piecewise quasi-polynomial.

Translating Theorem 1.10 into properties of S and χ_S, we have

$$1 \Rightarrow (2' \Leftrightarrow 3).$$

So we need to prove $2 \Rightarrow 1$ and $2' \Rightarrow 2$.

2 $\Rightarrow$ **1**.

This is straightforward: the property of being an element of $\lambda + \Lambda$ can be written using linear congruences and existential quantifiers, and the property of being an element of P can be written as a set of linear inequalities.

2′ $\Rightarrow$ **2**.

Since χ_S is a piecewise quasi-polynomial, it is constituted from associated polynomials. Let us examine such a polynomial $q(\mathbf{p})$ that agrees with χ_S on some $P \cap (\lambda + \Lambda)$, where $P \subseteq \mathbb{R}^n_{\geq 0}$ is a polyhedron, $\lambda \in \mathbb{Z}^n$, and Λ a sublattice of $\mathbb{Z}^n$. It suffices to prove that 2 holds for $S \cap P \cap (\lambda + \Lambda)$, since S is the disjoint union of such pieces.

Ideally, we would like to argue that, since q only takes on the values 0 and 1, the polynomial q must be constant on $P \cap (\lambda + \Lambda)$, at least if P is unbounded. This is not quite true; for example, if

$$P = \{(x, y) \in \mathbb{R}^2 : x \geq 0 \text{ and } 0 \leq y \leq 1\},$$

then the polynomial $q(x, y) = y$ is 1 for $y = 1$ and 0 for $y = 0$.

What we can say is that q must be constant on any infinite ray contained in $P \cap (\lambda + \Lambda)$: if we parametrize the ray by $\mathbf{x}(t) = (x_1(t), \cdots, x_n(t))$, then $q(\mathbf{x}(t))$ is a *univariate* polynomial that is either 0 or 1 at an infinite number of points, and so must be constant. Inductively, we can similarly show that q must be constant on any cone contained in P.

https://doi.org/10.1017/jsl.2015.4 Published online by Cambridge University Press

Let K be the cone with vertex at the origin

$$K = \{\mathbf{y} \in \mathbb{R}^n : \ \mathbf{y} + P \subseteq P\}.$$

Then K is the largest cone such that the cones $\mathbf{x} + K$ are contained in P, for all $\mathbf{x} \in P$; K is often called the *recession cone* or *characteristic cone* of P (see Section 8.2 of [47]), and the polyhedron P can be decomposed into a Minkowski sum $K + Q$, where Q is a *bounded* polyhedron. We can write $P \cap (\lambda + \Lambda)$ as a finite union (possibly with overlap) of sets of the form

$$Q_j = (v_j + K) \cap (\lambda + \Lambda),$$

for some v_j, and on each of these pieces q must be constant. If q is the constant 1 on Q_j, then Q_j is contained in S, and if q is the constant 0, then none of Q_j is in S. Since S is a finite union of the appropriate Q_j, S has the form needed for Property 2.

4.3. Proof of Proposition 1.12. Given that g is a piecewise quasi-polynomial with range in $\mathbb{N}$, we must find a Frobenius formula $F(\mathbf{c}, p)$ such that $g(p) = \#\{\mathbf{c} \in \mathbb{N}^d : F(\mathbf{c}, p)\}$. It suffices to find an F that agrees with g for sufficiently large p, because for any finite set $\{p_i\}$ we may include

$$(p = p_i) \Rightarrow \big(1 \leq c_1 \leq g(p_i) \wedge c_2 = \cdots = c_d = 0\big)$$

in the creation of F, so that the number of $\mathbf{c}$ satisfying F is exactly $g(p_i)$.

Since the domain of g is one-dimensional, a piecewise quasi-polynomial can be thought of as a function that is a quasi-polynomial for sufficiently large p. Therefore we may assume, without loss of generality, that g is a quasi-polynomial. Each lattice coset may be handled separately and combined to form the final formula, F. Therefore we may assume that g is a polynomial.

Let m be the least common multiple of the denominators of the coefficients of $g \in \mathbb{Q}[p]$. Suppose $p = mb + i$, for some b (which can be encoded in a Presburger formula, with m and i constants and b a bound variable). Considering $(mb + i)^k$ as a polynomial in b, all coefficients but (possibly) the constant term are multiples of m, so considering $g(mb + i)$ as a polynomial in b, all coefficients but (possibly) the constant term will be integral. Since $g(mb + i)$ is integer-valued, the constant term must also be an integer, that is, $g(mb + i) \in \mathbb{Z}[b]$. Since each residue class of p mod m can be handled separately and combined at the end, we may assume without loss of generality that $m = 1$ and $g \in \mathbb{Z}[p]$.

Let $g(p) = \sum_{i=0}^{k} a_i p^i$, $a_i \in \mathbb{Z}$. Since $g(p) \geq 0$ for all p, the leading coefficient, a_k, must be positive. Then $a_k p^k$ is the number of $\mathbf{c} \in \mathbb{N}^{k+1}$ such that

$$(1 \leq c_0 \leq a_k) \wedge (1 \leq c_1 \leq p) \wedge \cdots \wedge (1 \leq c_k \leq p).$$

All other *positive* terms of g can be similarly encoded as counting solutions to a Presburger formula (carefully defined so the solution sets are disjoint). To take care of a *negative* term $a_i p^i$, take p sufficiently large so that $a_k p^k$ sufficiently dominates

https://doi.org/10.1017/jsl.2015.4 Published online by Cambridge University Press

all other terms (as before, the finite set of smaller p can be dealt with separately); now remove $a_i p^i$ points from the set of solutions. For example,

$$\neg\Big((c_0 = c_{i+2} = \cdots = c_k = 1) \wedge (1 \le c_1 \le a_i) \\ \wedge (1 \le c_2 \le p) \wedge \cdots \wedge (1 \le c_{i+1} \le p)\Big)$$

removes $a_i p^i$ solutions, as long as $p \ge a_i$ (to do this for multiple i, define these subtractions carefully so that the solution sets do not overlap). All together, this give us our desired $F(\mathbf{c}, p)$ certifying that g is a Presburger counting function.

§5. Computational aspects. We first describe computational aspects of $A \Rightarrow (B \text{ and } C)$. Many partial results are known here, but there are several open problems. Then we discuss $B \Leftrightarrow C$, which works nicely: not only are B and C logically equivalent, but, after we make sense of the problem, they are computationally equivalent (in fixed dimension). Finally, we discuss other related complexity results.

5.1. $A \Rightarrow (B$ **and** $C)$**.** Note that another reasonable class of problems to study would be sentences where we are also allowed multiplication of variables, for example

$$\exists a \in \mathbb{N}, \exists b \in \mathbb{N} : \ a^2 + 2b^2 \le 31.$$

These problems, however, are very hard: undecidability, in the general case, is a consequence of Gödel's First Incompleteness Theorem [27]. Even if we focus on sentences with only existential quantifiers and a fixed number of variables (restrictions that we shall see work well for Presburger arithmetic), it is still undecidable: In fact, there is a certain multivariate polynomial $p(x_0, x_1, \ldots, x_d)$ such that the class of problems

$$\text{Given } a \in \mathbb{N}, \text{ decide whether } \exists b_1, \exists b_2, \ldots, \exists b_d : \\ p(a, b_1, b_2, \ldots, b_d) = 0, \text{ with } b_i \in \mathbb{N}$$

is undecidable. This is a consequence of the DPRM-theorem (after Davis, Putnam, Robinson, and Matiyasevich, see, for example, [19]), which solves Hilbert's 10th problem in the negative.

The Presburger arithmetic problem (deciding whether a Presburger sentence is true or false) is, at least, decidable, as originally proved [40] by Presburger in 1929. Since then, better algorithms have been found. For example, D. Oppen gave an algorithm [38], based on work of D. Cooper [17], with running time $2^{2^{2^{c\phi}}}$, where ϕ is the input size of the problem and c is a constant. Nevertheless, Fischer and Rabin gave lower bounds on the running time [23]: any algorithm that solves all Presburger arithmetic problems will sometimes take at least $2^{2^{c'\phi}}$ steps, where c' is a constant.

Among the simplest Presburger sentences are integer programming problems, which are themselves NP-complete if the dimension is not fixed. This suggests that any interesting polynomial time results will require a fixed number of variables.

https://doi.org/10.1017/jsl.2015.4 Published online by Cambridge University Press

We discuss three types of problems for a Presburger formula $F(\mathbf{u})$:

- The *decision* problem: Is $\exists \mathbf{u}\ F(\mathbf{u})$ true or false?
- The *counting* problem: How many solutions $\mathbf{u}$ to $F(\mathbf{u})$ are there?
- The *generating function* problem: Compute $\sum_{\mathbf{u}:\ F(\mathbf{u})} \mathbf{x}^{\mathbf{u}}$ as a rational generating function.

Clearly, the counting problem is harder than the decision problem. Now notice that if $\sum_{\mathbf{u}:\ F(\mathbf{u})} \mathbf{x}^{\mathbf{u}}$ can be computed in polynomial time, then the counting problem $\#\mathbf{u}\ F(\mathbf{u})$ can be solved in polynomial time: use Theorem 3.5 to plug in $\mathbf{x} = (1, 1, \ldots, 1)$. Consequently, the generating function problem is the hardest of the three and the decision problem $\exists \mathbf{u}\ F(\mathbf{u})$ is the easiest. Historically, polynomial time results have been discovered first for the decision problems, and these solutions have not needed generating functions. However, polynomial time results for counting problems have generally required the full power of generating functions. We now discuss several of these results.

First we examine, for a fixed number of variables and *quantifier-free* $F(\mathbf{u})$, deciding $\exists \mathbf{u}\ F(\mathbf{u})$. This is effectively Lenstra's algorithm for integer programming [35], which decides whether there exists a $\mathbf{u}$ satisfying the conjunction of several linear inequalities. The only additional step we need is to note that a Boolean combination of linear inequalities can be put in disjunctive normal form in polynomial time. This is slightly surprising, because it is not doable for general Boolean functions: for example, the disjunctive normal form for $\bigwedge_{i=1}^{n}(A_{i1} \vee A_{i2})$ requires 2^n disjunctions.

PROPOSITION 5.1. *Fix d. There is a polynomial time algorithm which, given a quantifier-free formula $F(u_1, u_2, \ldots, u_d)$ consisting of linear inequalities and Boolean operations, converts F into disjunctive normal form.*

PROOF. The inequalities appearing in F cut $\mathbb{R}^d$ into many polyhedral pieces. It might appear at first glance that the number of such pieces could be 2^N, where N is the total number of inequalities in F. Nevertheless, the number of pieces is bounded by

$$\Phi(d, N) = \binom{N}{0} + \binom{N}{1} + \cdots + \binom{N}{d}.$$

See Section 6.1 of [36] for a proof by induction. We have that $\Phi(d, N)$ is a polynomial in N of degree at most d (where d is fixed), and following the inductive proof, we see that the description of each piece (and the lower-dimensional polyhedral pieces on their boundaries) may be found in polynomial time. Within the interior of each piece, F is either always true or always false. Then the disjunctive normal form is simply $\bigvee_P \{x \in \text{int}(P)\}$, where the disjunction is taken over all polyhedral pieces P (including lower-dimensional ones) such that F is true. ⊣

The counting and generating function problems for quantifier-free Presburger formulas were proven to be polynomial time by Barvinok [1].

Adding one quantifier alternation makes this problem hard, unfortunately: even with two variables, deciding $\exists a \forall b\ F(a, b)$ is NP-hard, as Schöning showed [46]. In general, Grädel showed [28] that these problems rapidly increase in difficulty with increased quantifier alternation, following the polynomial-time hierarchy.

The above discussion demonstrates that, if one considers only fixing the number of quantifier alternations or number of variables, the tractability of Presburger

https://doi.org/10.1017/jsl.2015.4 Published online by Cambridge University Press

arithmetic is understood (and see [31], where Haase gives a few more variations). A possible modification is to also fix the number of linear inequalities allowed. In this case, Kannan proved [33] that the decision problem $\exists \mathbf{a} \forall \mathbf{b}\, F(\mathbf{a}, \mathbf{b})$ can be solved in polynomial time, and Barvinok and Woods proved [5] that the counting and generating function problems could be solved in polynomial time.

With more quantifier alternation, the computational complexity of the problem is unknown.

CONJECTURE 5.2. *For a fixed number of variables and linear inequalities, there exists a polynomial time algorithm to compute the generating function for a Presburger set. Hence the counting problem and the decision problem may be answered in polynomial time.*

5.2. $B \Leftrightarrow C$**.** The first question here is how to interpret the problem: how should a quasi-polynomial be represented? The obvious way, listing every constituent polynomial, is inefficient. For example, the generating function $1/(1-x^n)$ corresponds to the quasi-polynomial $g(x) = 1$ if $n \big| x$ and $g(x) = 0$ otherwise; this requires n polynomials to represent. The solution of Verdoolaege and Woods [54] is to use *piecewise step-polynomials*:

DEFINITION 5.3. A *step-polynomial* $g : \mathbb{N}^n \to \mathbb{Q}$ is a function written in the form

$$g(\mathbf{p}) = \sum_{i=1}^{m} \alpha_i \prod_{j=1}^{d_i} \lfloor a_{ij1} p_1 + \cdots + a_{ijn} p_n + b_{ij} \rfloor ,$$

where $\alpha_i, a_{ijk}, b_{ij} \in \mathbb{Q}$ and $\lfloor \cdot \rfloor$ is the greatest integer function.

For example, the counting function corresponding to $1/(1-x^n)$ can be written at the step-polynomial $\lfloor \frac{x}{n} \rfloor - \lfloor \frac{x-1}{n} \rfloor$.

DEFINITION 5.4. A *piecewise step-polynomial* is a function $g : \mathbb{N}^n \to \mathbb{Q}$ such that there exists a finite partition $\bigcup_i (P_i \cap \mathbb{N}^n)$ of $\mathbb{N}^n$ with P_i polyhedra (which may not all be full dimensional) and there exists step-polynomials g_i such that

$$g(\mathbf{p}) = g_i(\mathbf{p}) \text{ for } \mathbf{p} \in P_i \cap \mathbb{N}^n.$$

Verdoolaege and Woods prove [54] that one may convert between rational generating functions and piecewise step-polynomials in polynomial time (for fixed degree of the step-polynomial). Therefore, the conceptual equivalence, $B \Leftrightarrow C$, is also an algorithmic equivalence, provided piecewise quasi-polynomials are suitably represented.

5.3. Other Complexity Relationships. One reason generating functions are so valuable is that algebraic manipulations can be brought into play. We've already seen that substituting in 1 for variables can be done in polynomial time, enabling us to compute cardinality. Several other interesting operations have polynomial time algorithms.

For example, given the generating functions for two sets S and T, the generating functions for $S \cap T$, $S \cup T$, and $S \setminus T$ can all be computed in polynomial time [5] (assuming fixed number of variables and fixed bound on the number of binomials

https://doi.org/10.1017/jsl.2015.4 Published online by Cambridge University Press

in the denominators of the generating functions). This means that the Boolean operations and/or/not can be mirrored by generating functions. The key ingredient is the Hadamard product, which can be computed in polynomial time [5].

DEFINITION 5.5. The *Hadamard product* of

$$f(\mathbf{x}) = \sum_{\mathbf{p} \in \mathbb{N}^n} c(\mathbf{p})\mathbf{x}^{\mathbf{p}} \text{ and } g(\mathbf{x}) = \sum_{\mathbf{p} \in \mathbb{N}^n} d(\mathbf{p})\mathbf{x}^{\mathbf{p}}$$

is

$$(f \star g)(\mathbf{x}) = \sum_{\mathbf{p} \in \mathbb{N}^n} c(\mathbf{p})d(\mathbf{p})\mathbf{x}^{\mathbf{p}}.$$

For example, the fact that $f(S \cap T; \mathbf{x}) = f(S; \mathbf{x}) \star f(T; \mathbf{x})$ immediately shows us how to compute the generating function for $S \cap T$, given $f(S; \mathbf{x})$ and $f(T; \mathbf{x})$.

Unfortunately, the operation of quantification cannot be similarly mirrored efficiently with generating functions: if we are given $f(S; \mathbf{x}, y)$ for some $S \subseteq \mathbb{N}^{d+1}$, and define $T = \{\mathbf{a} \in \mathbb{N}^d : \forall b \in \mathbb{N}, (\mathbf{a}, b) \in S\}$, then obtaining $f(T; \mathbf{x})$ is NP-hard. To see this, take S to be the Presburger set for a quantifier free formula, $F(a, b)$. We can compute $f(S; x, y)$ in polynomial time, as discussed in Section 5.1. If we could compute $f(T; x)$ in polynomial time, then we would be able to decide $\exists a \forall b\, F(a, b)$ in polynomial time, by checking if $f(T; 1) \neq 0$. But deciding this is NP-hard [46], as mentioned in Section 5.1.

Even simpler sounding tasks can be surprisingly difficult. There is no presently-known "nice" way to tell whether a rational generating function, $g(\mathbf{x})$ (input as a sum of the basic rational functions from Definition 1.4), is identically zero. A heavy-handed – but polynomial time – algorithm is, after checking that $\mathbf{1}$ is not a pole of g, to substitute $\mathbf{1}$ into $g \star g$: the coefficients of $g \star g$ are nonnegative, so $(g \star g)(\mathbf{1}) = 0$ if and only if $g = 0$.

§6. Acknowledgments. My thanks to the referees for pointers to relevant related works. I am also grateful to them for helping me talk like a logician and like a theoretical computer scientist (when, in fact, I am a combinatorist).

REFERENCES

[1] ALEXANDER BARVINOK, *A polynomial time algorithm for counting integral points in polyhedra when the dimension is fixed*. ***Mathematics of Operations Research***, vol. 19 (1994), no. 4, pp. 769–779.

[2] ———, ***A Course in Convexity***, Graduate Studies in Mathematics, vol. 54, American Mathematical Society, Providence, RI, 2002.

[3] ———, *The complexity of generating functions for integer points in polyhedra and beyond*, ***International Congress of Mathematicians. Vol. III***, European Mathematical Society, Zürich, 2006, pp. 763–787.

[4] ALEXANDER BARVINOK and JAMES POMMERSHEIM, *An algorithmic theory of lattice points in polyhedra*, ***New perspectives in algebraic combinatorics (Berkeley, CA, 1996–97)***, Mathematical Sciences Research Institute Publications, vol. 38, Cambridge University Press, Cambridge, 1999, pp. 91–147.

[5] ALEXANDER BARVINOK and KEVIN WOODS, *Short rational generating functions for lattice point problems*. ***Journal of the American Mathematical Society***, vol. 16 (2003), no. 4, pp. 957–979.

https://doi.org/10.1017/jsl.2015.4 Published online by Cambridge University Press

[6] Matthias Beck, *The partial-fractions method for counting solutions to integral linear systems*. ***Discrete & Computational Geometry***, vol. 32 (2004), no. 4, pp. 437–446.

[7] Matthias Beck and Sinai Robins, ***Computing the continuous discretely***, Undergraduate Texts in Mathematics, Springer, New York, 2007.

[8] Leonard Berman, *The complexity of logical theories*. ***Theoretical Computer Science***, vol. 11 (1980), no. 1, pp. 57, 71–77.

[9] Víctor Blanco, Pedro A. García-Sánchez, and Justo Puerto, *Counting numerical semigroups with short generating functions*. ***International Journal of Algebra and Computation***, vol. 21 (2011), no. 7, pp. 1217–1235.

[10] Alexandre Boudet and Hubert Comon, *Diophantine equations, Presburger arithmetic and finite automata*, ***Trees in algebra and programming—CAAP '96 (Linköping, 1996)***, Lecture Notes in Computer Science, vol. 1059, Springer, Berlin, 1996, pp. 30–43.

[11] Michel Brion, *Points entiers dans les polyèdres convexes*. ***Annales scientifiques de l'École Normale Supérieure***, vol. 4 (1988), pp. 653–663.

[12] J. Richard Büchi, *Weak second-order arithmetic and finite automata*. ***Zeitschrift für mathematische Logik und Grundlagen der Mathematik***, vol. 6 (1960), pp. 66–92.

[13] J. W. S. Cassels, ***An introduction to the geometry of numbers***, Classics in Mathematics, Springer-Verlag, Berlin, 1997. Corrected reprint of the 1971 edition.

[14] Phillipe Clauss and Vincent Loechner, *Parametric analysis of polyhedral iteration spaces*. ***Journal of VLSI Signal Processing***, vol. 19 (1998), no. 2, pp. 179–194.

[15] Alan Cobham, *On the base-dependence of sets of numbers recognizable by finite automata*. ***Mathematical Systems Theory***, vol. 3 (1969), pp. 186–192.

[16] Hubert Comon and Yan Jurski, *Multiple counters automata, safety analysis and Presburger arithmetic*, ***Computer aided verification (Vancouver, BC, 1998)***, Lecture Notes in Computer Science, vol. 1427, Springer, Berlin, 1998, pp. 268–279.

[17] D.C. Cooper, *Theorem proving in arithmetic without multiplication*. ***Machine Intelligence***, vol. 7 (1972), pp. 91–99.

[18] Flavio D'alessandro, Benedetto Intrigila, and Stefano Varricchio, *Quasi-polynomials, linear Diophantine equations and semi-linear sets.* ***Theoretical Computer Science***, vol. 416 (2012), pp. 1–16.

[19] Martin Davis, *Hilbert's tenth problem is unsolvable*. ***American Mathematical Monthly***, vol. 80 (1973), pp. 233–269.

[20] Jesus De Loera, David Haws, Raymond Hemmecke, Peter Huggins, Bernd Sturmfels, and Ruriko Yoshida, *Short rational functions for toric algebra and applications.* ***Journal of Symbolic Computation***, vol. 38 (2004), no. 2, pp. 959–973.

[21] Eugène Ehrhart, *Sur les polyèdres rationnels homothétiques à n dimensions*. ***Comptes rendus de l'Académie des sciences Paris***, vol. 254 (1962), pp. 616–618.

[22] Jeanne Ferrante and Charles W. Rackoff, ***The computational complexity of logical theories***, Lecture Notes in Mathematics, vol. 718, Springer, Berlin, 1979.

[23] Michael Fischer and Michael Rabin, *Super-exponential complexity of Presburger arithmetic*, ***Complexity of computation (Proceedings of SIAM-AMS symposium, New York, 1973)***, American Mathematical Society, Providence, R.I, 1974, pp. 27–41. SIAM-AMS Proceedings, Vol. VII.

[24] William Fulton, ***Introduction to Toric Varieties***, Annals of Mathematics Studies, vol. 131, Princeton University Press, Princeton, NJ, 1993.

[25] Martin Fürer, *The complexity of Presburger arithmetic with bounded quantifier alternation depth*. ***Theoretical Computer Science***, vol. 18 (1982), no. 1, pp. 105–111.

[26] Seymour Ginsburg and Edwin Spanier, *Semigroups, Presburger formulas and languages*. ***Pacific Journal of Mathematics***, vol. 16 (1966), no. 2, pp. 285–296.

[27] Kurt Gödel, *Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme. I*. ***Monatshefte für Mathematik***, vol. 149 (2006), no. 1, pp. 1–30. Reprinted from ***Monatshefte für Mathematik und Physik***. **3**8 (1931), 173–198. With an introduction by Sy-David Friedman.

[28] Erich Grädel, *Subclasses of Presburger arithmetic and the polynomial-time hierarchy*. ***Theoretical Computer Science***, vol. 56 (1988), no. 3, pp. 289–301.

[29] ———, *Dominoes and the complexity of subclasses of logical theories*. ***Annals of Pure and Applied Logic***, vol. 43 (1989), no. 1, pp. 1–30.

https://doi.org/10.1017/jsl.2015.4 Published online by Cambridge University Press

[30] Alan Guo and Ezra Miller, *Lattice point methods for combinatorial games*. ***Advances in Applied Mathematics***, vol. 46 (2011), no. 1–4, pp. 363–378.

[31] Christoph Haase, ***Subclasses of Presburger Arithmetic and the Weak EXP Hierarchy***, Proceedings of the Joint Meeting of the 23rd EACSL Annual Conference on Computer Science Logic and the 29th Annual ACM/IEEE Symposium on Logic in Computer Science (CSL/LICS'14), ACM Press, Vienna, Austria, 2014.

[32] Serkan Hoşten and Bernd Sturmfels, *Computing the integer programming gap.* ***Combinatorica. An International Journal on Combinatorics and the Theory of Computing***, vol. 27 (2007), no. 3, pp. 367–382.

[33] Ravi Kannan, *Test sets for integer programs, ∀∃ sentences*, ***Polyhedral combinatorics (Morristown, NJ, 1989)***, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, vol. 1, American Mathematical Society, Providence, RI, 1990, pp. 39–47.

[34] Felix Klaedtke, *Bounds on the automata size for Presburger arithmetic*. ***ACM Transactions on Computational Logic***, vol. 9 (2008), no. 2, pp. Art. 11, 34.

[35] Hendrik Lenstra, Jr., *Integer programming with a fixed number of variables*. ***Mathematics of Operations Research***, vol. 8 (1983), no. 4, pp. 538–548.

[36] Jiri Matousek, ***Lectures on Discrete Geometry***, Graduate Texts in Mathematics, vol. 212, Springer-Verlag, New York, 2002.

[37] Ezra Miller and Bernd Sturmfels, ***Combinatorial commutative algebra***, Graduate Texts in Mathematics, vol. 227, Springer-Verlag, New York, 2005.

[38] Derek Oppen, *A superexponential upper bound on the complexity of Presburger arithmetic*. ***Journal of Computer and System Sciences***, vol. 16 (1978), no. 3, pp. 323–332.

[39] Erin Parker and Siddhartha Chatterjee, *An automata-theoretic algorithm for counting solutions to Presburger formulas*, ***Compiler construction***, Springer, 2004, pp. 104–119.

[40] Mojżesz Presburger, *On the completeness of a certain system of arithmetic of whole numbers in which addition occurs as the only operation*, ***History and Philosophy of Logic***, vol. 12 (1991), no. 2, pp. 225–233. Translated from the German and with commentaries by Dale Jacquette.

[41] William Pugh, *Counting solutions to Presburger formulas: how and why*. ***SIGPLAN Notices***, vol. 29 (1994), no. 6, pp. 121–134.

[42] J. L. Ramírez Alfonsín, ***The Diophantine Frobenius problem***, Oxford Lecture Series in Mathematics and its Applications, vol. 30, Oxford University Press, Oxford, 2005.

[43] Cattamanchi R Reddy and Donald W Loveland, *Presburger arithmetic with bounded quantifier alternation*. ***Proceedings of the Tenth Annual ACM Symposium on Theory of Computing***, ACM, 1978, pp. 320–325.

[44] Herbert Scarf, *Test sets for integer programs*. ***Mathematical Programming***, vol. 79 (1997), no. 1–3, Series B, pp. 355–368.

[45] Bruno Scarpellini, *Complexity of subcases of Presburger arithmetic*. ***Transactions of the American Mathematical Society***, vol. 284 (1984), no. 1, pp. 203–218.

[46] Uwe Schöning, *Complexity of Presburger arithmetic with fixed quantifier dimension*. ***Theory of Computing Systems***, vol. 30 (1997), no. 4, pp. 423–428.

[47] Alexander Schrijver, ***Theory of Linear and Integer Programming***, Wiley-Interscience Series in Discrete Mathematics, John Wiley & Sons Ltd., Chichester, 1986.

[48] ———, ***Combinatorial optimization. Polyhedra and efficiency***, Algorithms and Combinatorics, vol. 24, Springer-Verlag, Berlin, 2003.

[49] Richard P. Stanley, *Decompositions of rational convex polytopes*, ***Annals of Discrete Mathematics***, vol. 6 (1980), pp. 333–342.

[50] Bernd Sturmfels, *On vector partition functions*. ***Journal of Combinatorial Theory, Series A***, vol. 72 (1995), no. 2, pp. 302–309.

[51] ———, ***Gröbner Bases and Convex Polytopes***, University Lecture Series, vol. 8, American Mathematical Society, Providence, RI, 1996.

[52] Rekha Thomas, *A geometric Buchberger algorithm for integer programming*. ***Mathematics of Operations Research***, vol. 20 (1995), no. 4, pp. 864–884.

[53] ———, ***The structure of group relaxations***, Handbooks in Operations Research and Management Science, vol. 12, Elsevier, Amsterdam, 2005, pp. 123–170.

[54] Sven Verdoolaege and Kevin Woods, *Counting with rational generating functions*. ***Journal of Symbolic Computation***, vol. 43 (2008), no. 2, pp. 75–91.

https://doi.org/10.1017/jsl.2015.4 Published online by Cambridge University Press

[55] Pierre Wolper and Bernard Boigelot, *An automata-theoretic approach to Presburger arithmetic constraints*, ***Static analysis, 2nd international symposium***, Lecture Notes in Computer Science, vol. 983, Springer, Berlin, 1995, pp. 21–32.

[56] Kevin Woods, ***Rational generating functions and lattice point sets***, Ph.D. thesis, University of Michigan, Ann Arbor, 2004.

DEPARTMENT OF MATHEMATICS
OBERLIN COLLEGE
OBERLIN, OHIO 44074, USA
E-mail: kevin.woods@oberlin.edu
URL: http://www.oberlin.edu/faculty/kwoods/

https://doi.org/10.1017/jsl.2015.4 Published online by Cambridge University Press