Integer Hulls, $\mathbb{Z}$-Polyhedra and Presburger Arithmetic in Action

Rui-Juan Jing[1], Yuzhuo Lei[2], Christopher F. S. Maligec[2], Marc Moreno Maza[2], and Chirantan Mukherjee[2]

[1]School of Mathematical Science, Jiangsu University, Zhenjiang, China
rjing@ujs.edu.cn
[2]Ontario Research Center for Computer Algebra, University of Western Ontario, London, Ontario, Canada
{ylei83,cmaligec,mmorenom,cmukher}@uwo.ca

June 10, 2025

We recommend viewing this article with Adobe Reader or Okular in order to see the animations.

1 Overview

When solving systems of polynomial equations and inequalities, the task of computing their solutions with integer coordinates is a much harder problem than that of computing their real solutions or that of computing all their solutions. In fact, in the presence of non-linear constraints, this task may simply become an undecidable problem [12, 15]. However, studying the integer solutions of linear systems of equations and inequalities is of practical importance in various areas of scientific computing. Two such areas are *combinatorial optimization* (in particular, integer linear programming) and *compiler optimization* (in particular, the analysis, transformation, and scheduling of nested loops in computer programs), where a variety of algorithms solve questions related to the points with integer coordinates in a given polyhedron. Another area is at the crossroads of computer algebra and polyhedral geometry, with topics such as toric ideals and Hilbert bases, see [16], as well as the manipulation of Laurent series, see [1].

There are different problems regarding the integer points of a polyhedral set, ranging from whether or not a given rational polyhedron has integer points to "describing all such points." Answers to the latter can take various forms, depending on the targeted application. For plotting purposes, one may want to enumerate all the integer points of a 2D or 3D polytope, whereas, in the context of combinatorial optimization or compiler optimization, more concise descriptions are sufficient and more effective. For a rational convex polyhedron $P \subseteq \mathbb{Q}^d$, defined either by the set of its facets or that of its vertices, one such description is the *integer hull* P_I of P, that is, the convex hull of $P \cap \mathbb{Z}^d$. The set P_I is itself polyhedral and can be described either by its facets, or its vertices.

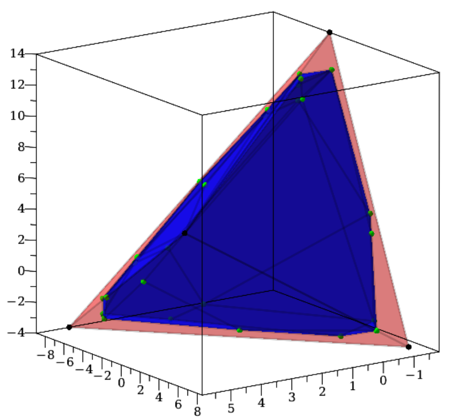
Another concise description was proposed in [10, 11], where the integer points of a polyhedral set are represented by (finitely many) triangular systems with specialization properties similar to those of regular chains and lexicographical Gröbner bases.

An even more concise answer is given by counting the number of integer points of a polytope. This problem has efficient algorithmic solutions, in particular, Barvinok's algorithm [2], as well as numerous applications, such as counting the number of memory locations (or cache lines) accessed by a for-loop nest, see [14, 4].

Over the past twelve years, a series of projects has equipped the computer algebra system MAPLE with a number of tools for dealing with the integer points of systems of linear equations and inequalities, even in the presence of parameters. These tools implement novel and effective algorithms; they are part of the `PolyhedralSets` and `QuantifierEliminationOverZ` libraries, and they are the main focus of the proposed software demo. See [9] and [8] for applications of these tools.

2 Describing or counting the integer points of polyhedral sets

With some of the commands of the `PolyhedralSets` library, one can either decide whether a polyhedral set has integer points or simply count them. One can also describe these points in a compact way or enumerate them. Last but not least, one can deal with parametric polytopes and count the number of their integer points.



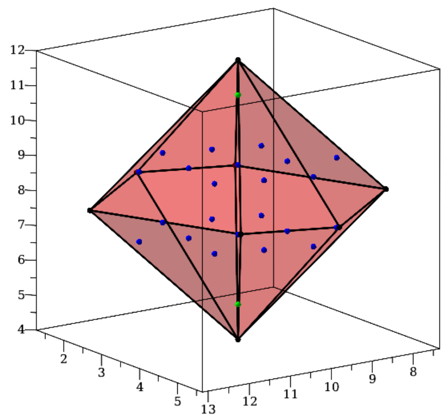
We start with a non-parametric polytope, the tetrahedron P shown in red on Figure 1. Its integer hull P_I is plotted in blue. The integer points of P, other than the vertices of P_I, are shown in green. The vertices of the tetrahedron P are given by $v = \left\{(4,8,6),\ (-1,-\frac{10}{3},14),\ (-3,8,-\frac{10}{3}),\ (7,-\frac{22}{3},-4)\right\}$, from which we obtain the inequalities defining P:

$$-x_1 - \frac{131}{205}x_2 - \frac{62}{205}x_3 \le -\frac{679}{615}, \quad -x_1 + \frac{33}{34}x_2 + \frac{3}{4}x_3 \le \frac{281}{34},$$
$$x_1 - \frac{13}{118}x_2 + \frac{83}{177}x_3 \le \frac{350}{59}, \qquad x_1 + \frac{63}{92}x_2 - \frac{3}{4}x_3 \le \frac{229}{46}$$

The `IntegerHull` command returns the 27 vertices of P_I.

Figure 1: A tetrahedron and its integer hull.

We continue with another polytope shown on Figure 2. It is given by the 12 inequalities defining its facets. We are interested in describing all its integer points, not just the vertices of its integer hull. The algorithm [10, 11], which is based on William Pugh's Omega test [13], performs this task.



In the ZPolyhedralSets sub-package of the PolyhedralSets library, this algorithm is implemented as the IntegerPointDecomposition command. For any given semi-linear set S, this command computes a partition of S into $\mathbb{Z}$-polyhedra, where a $\mathbb{Z}$-polyhedron is the intersection of a polyhedral set with an integer lattice. The polyhedron is defined by the inequalities:

$$
\begin{array}{ll}
0 \leq -16 + 2y + z, & 0 \leq -72 + 4x + 4y + 3z, \\
0 \leq 2y - z, & 0 \leq -24 + 4x + 4y - 3z, \\
0 \leq -4x + 4y + 3z, & 0 \leq 48 - 4x + 4y - 3z, \\
0 \leq 48 - 4x - 4y + 3z, & 0 \leq 8 - 2y + z, \\
0 \leq -24 + 4x - 4y + 3z, & 0 \leq 24 - 2y - z, \\
0 \leq 24 + 4x - 4y - 3z, & 0 \leq 96 - 4x - 4y - 3z
\end{array}
$$

Figure 2: A polyhedron (in red), its dark shadow (in blue), and grey shadow (in green) in the projection on the $y - z$ plane.

Relations : $\left\{\begin{array}{l} z \leq 9 \\ -z \leq -7 \\ -2y - z \leq -16 \\ -2y + z \leq 0 \\ 2y - z \leq 8 \\ 2y + z \leq 24 \\ -4x - 4y - 3z \leq -72 \\ -4x - 4y + 3z \leq -24 \\ -4x + 4y - 3z \leq -24 \\ -4x + 4y + 3z \leq 24 \\ 4x - 4y - 3z \leq 0 \\ 4x - 4y + 3z \leq 48 \\ 4x + 4y - 3z \leq 48 \\ 4x + 4y + 3z \leq 96 \end{array}\right.$

Variables : $[x, y, z]$

Parameters : [ ]

ParameterConstraints : {

Lattice : $\mathrm{ZSpan}\left(\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}\right)$

Relations : $\left\{\begin{array}{l} x = 9 \\ y = 6 \\ z = 5 \end{array}\right.$

Variables : $[x, y, z]$

Parameters : [ ]

ParameterConstraints : {

Lattice : $\mathrm{ZSpan}\left(\begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 2 \end{bmatrix}, \begin{bmatrix} 9 \\ 0 \\ 1 \end{bmatrix}\right)$

Relations : $\left\{\begin{array}{l} x = 9 \\ y = 6 \\ z = 11 \end{array}\right.$

Variables : $[x, y, z]$

Parameters : [ ]

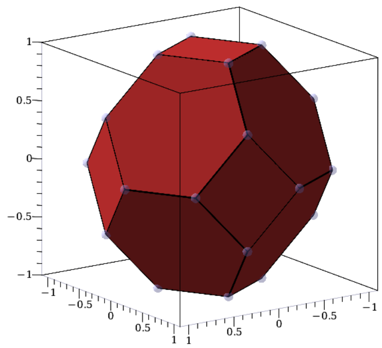
ParameterConstraints : {

Lattice : $\mathrm{ZSpan}\left(\begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 2 \end{bmatrix}, \begin{bmatrix} 9 \\ 0 \\ 1 \end{bmatrix}\right)$

Figure 3 shows the output of IntegerPointDecomposition for the integer hull of the polyhedra set P of Figure 2. This decomposition counts three $\mathbb{Z}$-polyhedra: the first one corresponds to the dark shadow (in the sense of William Pugh) of P in its projection on the $y - z$ plane, meanwhile the other two $\mathbb{Z}$-polyhedra form the grey shadow of P.

Figure 3: Applying IntegerPointDecomposition to the polytope of Figure 2 produces three $\mathbb{Z}$-polyhedra.

We turn our attention to the problem of counting the integer points of a polytope. Figure 4 shows a dodecahedron and its generating function (in the sense of Michel Brion [3]) computed by the command Generatingfunction of the PolyhedralSets library.



Once all coordinates are specialized to 1, this generating function becomes the number of integer points of the input polyhedron. To obtain this latter result, one can directly call the command `NumberOfIntegerPoints`, which also supports parametric polyhedral sets.

$gps := GeneratingFunction(ps);$

$$gps := \frac{x_1^2\, x_2\, x_3 + x_1\, x_2^2\, x_3 + x_1\, x_2\, x_3^2 + x_1\, x_2\, x_3 + x_1\, x_2 + x_1\, x_3 + x_2\, x_3}{x_3\, x_2\, x_1}$$

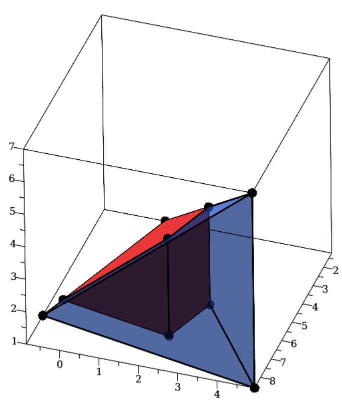
$eval(gps, [seq(v = 1, v \textbf{ in } Coordinates(ps))]);$

$$7$$

$NumberOfIntegerPoints(ps);$

$$7$$

Figure 4: A dodecahedron and its generating function.



We illustrate the fact that dealing with parametric polyhedral sets makes calculations substantially more complex. Indeed, Figure 5 shows a parametric polyhedron P, for which two instances have different numbers of vertices and facets. This observation helps understand why case discussions occur when counting the number of integer points of a parametric polyhedron. Figure 6 shows the number of integer points of P as a piece-wise function.
A parametric polyhedron P, with parameters n, m and coordinates i, j, k given by its defining inequalities: $1 \le i \le n, \quad 1 \le j \le m, \quad j \le i, \quad 1 \le k \le j.$

Figure 5: A parametric polyhedron and two of its instances: in red when $(m, n) = (4, 6)$ and in blue when $(m, n) = (9, 7)$.

$np := NumberOfIntegerPoints(parametric\_polytope\_2, [i, j, k], [m, n], output = piecewise);$

$$np := \begin{cases} \left\{\frac{1}{3}\, n + \frac{1}{2}\, n^2 + \frac{1}{6}\, n^3\right\} & \mathrm{And}(-m + n = 0, 0 \le -2 + n) \\ \{1\} & \mathrm{And}(m - 1 = 0, n - 1 = 0) \\ \left\{\frac{1}{3}\, n + \frac{1}{2}\, n^2 + \frac{1}{6}\, n^3\right\} & \mathrm{And}(0 \le -2 + n, 0 \le m - n - 1) \\ \{1\} & \mathrm{And}(n - 1 = 0, 0 \le m - 2) \\ \left\{\frac{1}{3}\, m + \frac{1}{2}\, m\, n + \frac{1}{2}\, n\, m^2 - \frac{1}{3}\, m^3\right\} & \mathrm{And}(0 \le m - 2, 0 \le n - 3, 0 \le -m + n - 1) \\ \{4 + n\} & \mathrm{And}(m - 1 = 0, 0 \le -2 + n) \end{cases}$$

Figure 6: The parametric polyhedron of Figure 5 and the number of its integer points as a piece-wise function.

The details about the underlying algorithms and the implementation of `NumberOfIntegerPoints` are reported in [8]. One remarkable aspect of the theory of parametric polyhedral sets is the fact that the numbers of their integer points can be given by piece-wise periodic functions, see the work of Eugène Ehrhart [5, 6]. These periodic functions are captured by the notion of *quasi-polynomials*. Figure 7 shows this for the parametric polyhedron $1 \le i,\ j \le n,\ i \le m,\ 3\,i \le 5\,j$, with parameters n, m and coordinates i, j. The number of its integer points is a piece-wise periodic function, using quasi-polynomials.

```
NumberOfIntegerPoints(parametric_polytope , [i, j], [m,n], output = piecewise, compactdisplay);
```

$$\begin{cases} \left\{-\frac{3\,m^2}{10} + \mathit{QuasiPolynomial}\left(\left[0, 0, -\frac{2}{5}, -\frac{1}{5}, -\frac{2}{5}\right], m, 5\right) + n\,m + \frac{3\,m}{10}\right\} & 0 \le m-2 \wedge 0 \le 5\,n-7 \wedge 0 \le 5\,n-3\,m-1 \\ \{n\} & m-1=0 \wedge 0 \le 5\,n-4 \\ \left\{\mathit{QuasiPolynomial}\left(\left[0, -\frac{1}{3}, -\frac{1}{3}\right], n, 3\right) + \frac{5\,n^2}{6} + \frac{n}{2}\right\} & 3\,m-5\,n=0 \wedge 0 \le 5\,n-4 \\ \left\{\mathit{QuasiPolynomial}\left(\left[0, -\frac{1}{3}, -\frac{1}{3}\right], n, 3\right) + \frac{5\,n^2}{6} + \frac{n}{2}\right\} & 0 \le 5\,n-4 \wedge 0 \le 3\,m-5\,n-1 \end{cases}$$

Figure 7: A parametric polyhedron and the number of its integer points, using quasi-polynomials.

3 Presburger arithmetic with `QuantifierEliminationOverZ`

The language of ordinary Presburger arithmetic is the first-order theory of the integers with addition extended by the divisibility predicates $D_k : x \longmapsto k \mid x$, for every positive integer k. In 1929, Mojżesz Presburger proved that this language admits quantifier elimination. See Christoph Haase's *Survival Guide to Presburger Arithmetic* [7] for an introduction to the subject.

Consider a Presburger formula F in prenex normal form: $F = Q_1 x_1 \cdots Q_m x_m\, \phi(x_1, \ldots, x_m, y_1, \ldots, y_n)$, where $Q_1 x_1 \cdots Q_m x_m$ is a sequence of quantifiers (existential or universal) and bound variables, followed by a quantifier-free part $\phi(x_1, \ldots, x_m, y_1, \ldots, y_n)$, where $y_1, \ldots, y_n$ are free (or unbounded) variables. Our goal is to determine the *domain* of F, that is, the set $D(y_1, \ldots, y_n)$ of the integer tuples of $(y_1, \ldots, y_n)$ for which the formula $F(x_1, \ldots, x_m, y_1, \ldots, y_n)$ is true. The command `QEoverZ` of the library `QuantifierEliminationOverZ` performs this task. The underlying algorithms are presented in [9], and our code for them is publicly available **here.** The other software packages mentioned in this extended abstract are shipped with MAPLE 2025.

We consider four illustrative examples. Each of them is given as a quantified formula `f, g, h, i`.

1. `f` $:= \forall x_1, \forall x_2, \exists y (x_1 + 1 < x_2) \Rightarrow (x_1 < y \wedge y < x_2)$,
2. `g` $:= \exists a\, (0 \le x \wedge x \le 100) \vee (5 \le x \wedge x \le 40 \wedge x = 5a)$,
3. `h` $:= \exists i,\ \exists j \{-m \le i - j,\ i + j \le n,\ p \le j\}$,
4. `i` $:= \exists x_1,\ \exists x_2 \{c \le 3x_1 + 4x_2,\ 3x_1 + 2x_2 \le 500v,\ 0 \le x_1 \le 100v,\ 0 \le x_2 \le 200v\}$.

Figure 8 shows a MAPLE session where the command `QEoverZ` is applied to the above four Presburger formulas. In particular, we see that this command returns `true` for the first formula. We see that `QEoverZ` returns the following for the second formula: $(0 \le x \le 100) \vee ((5 \le x \le 40, x \equiv 0 \mod 5)$, that is, the union of two $\mathbb{Z}$-polyhedra. For the third formula, the returned inequality is a necessary and sufficient constraint on the parameters m, n, p for the input parametric polyhedral set to have integer points.

```
> f := &A([x1]), &A([x2]), &E([y]), &implies(x1 + 1 < x2, &and(x1 < y, y < x2)) : QEoverZ(f);
```

$$\mathit{true}$$

```
> g := &E([a]), &or(&and(0 ≤ x, x ≤ 100), &and(5 ≤ x, x ≤ 40, x = 5*a)) : QEoverZ(g) ;
```

$$\mathit{Logic{:}\text{-}\&or}\Big(\mathit{Logic{:}\text{-}\&and}(x \le 100, -x \le 0), \mathit{Logic{:}\text{-}\&and}\Big([x] \in \mathit{PolyhedralSets{:}\text{-}ZPolyhedralSets{:}\text{-}Lattice}\Big(\big[\,5\,\big], \big[\,0\,\big]\Big), [x \le 40, -x \le -5]\Big)\Big)$$

```
> h := &E([i]),&E([j]), [i − j ≥ − m, i + j ≤ n,  p ≤ j] : QEoverZ(h);
```

$$[2\,p - m - n \le 0]$$

```
> i := &E([x1]), &E([x2]),  &and(c ≤ 3*x1 + 4*x2, 3*x1 + 2*x2 ≤ 500*v, −x1 ≤ 0, x1 ≤ 100*v, −x2 ≤ 0, x2 ≤ 200*v) :
      QEoverZ(i);
```

$$\mathit{Logic{:}\text{-}\&or}\left(\mathit{Logic{:}\text{-}\&and}\left([c, v] \in \mathit{PolyhedralSets{:}\text{-}ZPolyhedralSets{:}\text{-}Lattice}\left(\begin{bmatrix} 6 & 4 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix}\right), [-v \le 0, c - 900\,v \le 0, 700\,v - c \le 0]\right), \mathit{Logic{:}\text{-}\&and}\left([c, v] \in \mathit{PolyhedralSets{:}\text{-}ZPolyhedralSets{:}\text{-}Lattice}\left(\begin{bmatrix} 6 & 4 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}\right), [-v \le -1, c - 900\,v \le -1, -1000\,v + c \le -2, 700\,v - c \le 1]\right), \mathit{Logic{:}\text{-}\&and}\left([c, v] \in \mathit{PolyhedralSets{:}\text{-}ZPolyhedralSets{:}\text{-}Lattice}\left(\begin{bmatrix} 6 & 4 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 5 \\ 0 \end{bmatrix}\right), [-v \le 0, c - 900\,v \le -1, 700\,v - c \le 1]\right), \mathit{Logic{:}\text{-}\&and}(-v \le 0, c - 900\,v \le -2)\right)$$

Figure 8: Eliminating quantifiers for four Presburger formulas.

References

[1] A. Aparicio Monforte and M. Kauers. Formal Laurent series in several variables. *Expositiones Mathematicae*, 31(4):350–367, 2013.

[2] A. I. Barvinok. A polynomial time algorithm for counting integral points in polyhedra when the dimension is fixed. *Mathematics of Operations Research*, 19(4):769–779, 1994.

[3] M. Brion. Points entiers dans les polyedres convexes. In *Annales scientifiques de l'École normale supérieure*, volume 21, pages 653–663, 1988.

[4] P. Clauss, F. J. Fernández, D. Garbervetsky, and S. Verdoolaege. Symbolic polynomial maximization over convex sets and its application to memory requirement estimation. *IEEE Trans.*, 17(8):983–996, 2009.

[5] E. Ehrhart. Sur un problème de géométrie diophantienne linéaire. i. *Journal für die reine und angewandte Mathematik*, 226:1–29, 1967.

[6] E. Ehrhart. Polynômes arithmétiques et méthode des polyèdres en combinatoire. *International Series of Numerical Mathematics, Birkhäuser Verlag, Basel*, 35, 1977.

[7] C. Haase. A survival guide to presburger arithmetic. *ACM SIGLOG News*, 5(3):67–82, 2018.

[8] R. Jing, Y. Lei, C. F. S. Maligec, and M. Moreno Maza. Counting the integer points of parametric polytopes: A maple implementation. In *CASC 2024, Proceedings*, volume 14938 of *LNCS*, pages 140–160. Springer, 2024.

[9] R. Jing, Y. Lei, C. F. S. Maligec, M. Moreno Maza, and C. Mukherjee. Quantifier elimination over the integers. In *ISSAC 2025, Proceedings*. ACM, 2025. To appear.

[10] R. Jing and M. Moreno Maza. Computing the integer points of a polyhedron, I: algorithm. In *CASC 2017, Proceedings*, volume 10490 of *LNCS*, pages 225–241. Springer, 2017.

[11] R. Jing and M. Moreno Maza. Computing the integer points of a polyhedron, II: complexity estimates. In *CASC 2017, Proceedings*, volume 10490 of *LNCS*, pages 242–256. Springer, 2017.

[12] Y. Matiyasevivc. Enumerable sets are diophantine. *Mathematical logic in the 20th century*, pages 269–273, 2003.

[13] W. W. Pugh. The omega test: a fast and practical integer programming algorithm for dependence analysis. In *Proceedings Supercomputing 1991*, pages 4–13. ACM, 1991.

[14] W. W. Pugh. Counting solutions to presburger formulas: How and why. In *Proceedings of the ACM SIGPLAN 1994*, pages 121–134. ACM, 1994.

[15] J. Robinson. Unsolvable diophantine problems. *Proceedings of the AMS*, 22(2):534–538, 1969.

[16] R. R. Thomas. Integer programming: Algebraic methods. In *Encyclopedia of Optimization, Second Edition*, pages 1624–1634. Springer, 2009.