hexagon:2611.00678v1

ALGORITHMS FOR COMPUTING ISO-DELAUNAY DOMAINS

MATHIEU DUTOUR SIKIRIĆ

Abstract. For a linear space T of quadratic forms, we describe an algorithm enumerating the iso-Delaunay domains of T, and a randomized search for an iso-Delaunay domain all of whose extreme rays are positive definite. For the quadratic forms in six variables, the search finds 1472 pairwise inequivalent L-type domains all of whose extreme rays are positive definite, the smallest having 24 facets and 47 extreme rays. No such domain exists in dimension $2 \leq n \leq 5$.

1. Introduction

Let T be a linear space of symmetric $n \times n$ matrices containing a positive definite one. The positive definite forms of T that have the same Delaunay tessellation form a relatively open polyhedral cone, an iso-Delaunay domain, also called a T-secondary cone [16]. For $T = \mathrm{Sym}_n(\mathbb{R})$ these are the L-type domains of Voronoi's second reduction theory [24], and each full-dimensional one determines a combinatorial type of primitive parallelohedra.

Several programs enumerate iso-Delaunay domains. Some treat only the full space $T = \mathrm{Sym}_n(\mathbb{R})$: the secondary cone cruiser `scc` of Schürmann and Vallentin [22], which was one of the implementations used for the complete classification in dimension 5 [14], and the program of Baburin and Engel, who found 567 613 632 combinatorial types of primitive parallelohedra in dimension 6, a lower bound since their enumeration is not complete [3]. Others treat a general T-space: the GAP package `polyhedral` [9] and its C++ version [10], which is the one described and used here, and the Julia code of Miebach [18], who used it for the sphere coverings of Eisenstein, Gaussian and Hurwitzian lattices.

We consider the following question: is there a full-dimensional iso-Delaunay domain all of whose extreme rays are positive definite forms? The extreme rays of the L-type domains include the rigid forms, whose L-type domain is a single ray, and the question was raised in the study of the six-dimensional rigid forms [8]. In the classical case $T = \mathrm{Sym}_n(\mathbb{R})$ the extreme rays of an L-type domain often include forms of rank 1, and for $2 \leq n \leq 5$ every full-dimensional domain has such a ray. We show that the answer is positive for $n = 6$.

Theorem 1. *There exist L-type domains of positive definite quadratic forms in six variables, of dimension 21, all of whose extreme rays are positive definite.*

In Section 2 we describe the enumeration of the iso-Delaunay domains of a space T, together with a canonical interior point of a polyhedral cone, which is of independent interest. Section 3 gives test cases for the enumeration, over the Hermitian forms of imaginary quadratic rings. In Section 4 we describe a randomized search for a domain all of whose extreme rays have full rank, and in Section 5 we report its outcome for $n = 6$: 1472 pairwise inequivalent such domains.

2. Enumeration of iso-Delaunay domains

2.1. **The input space.** Let $T \subset \mathrm{Sym}_n(\mathbb{R})$ be the rational linear space of symmetric matrices under consideration. We use its saturated integral matrix lattice

$$T_{\mathbb{Z}} = T \cap \mathrm{Sym}_n(\mathbb{Z}) \tag{1}$$

and assume that it spans T over $\mathbb{R}$. Fix a $\mathbb{Z}$-basis $B_1, \dots, B_m$ of $T_{\mathbb{Z}}$, and write $\ell_u(x) = x^t B_u x$. Thus the integral coordinate vectors are exactly the integral matrices in T. A point $q = (q_1, \dots, q_m) \in \mathbb{R}^m$ represents the Gram matrix

$$Q(q) = \sum_{u=1}^{m} q_u B_u. \tag{2}$$

The enumeration is performed inside the positive definite part of T. A domain is represented by a Delaunay tessellation $\mathrm{Del}(Q)$, an interior Gram matrix Q, and an invariant short-vector family used for equivalence tests.

Definition 1. *An iso-Delaunay domain in T is the relatively open cone of all positive-definite forms $Q \in T$ that have a fixed Delaunay tessellation. Its closure is a rational polyhedral cone in T.*

In particular, if $U \in \mathrm{GL}_n(\mathbb{Z})$ preserves T, then $Q \mapsto U^t Q U$ preserves $T_{\mathbb{Z}}$. Its induced matrix in the above T-coordinates therefore belongs to $\mathrm{GL}_m(\mathbb{Z})$. We write $\mathrm{GL}(T_{\mathbb{Z}})$ for the group of these induced matrices. This integral coordinate structure is also used in the enumeration of T-perfect forms [21].

The algorithm follows the T-secondary-cone theory of [16], itself a generalization of Voronoi's reduction theory [24]. The finite object that is enumerated is not the full infinite tessellation of $\mathbb{R}^n$, but a list of orbits of Delaunay polytopes together with facet adjacencies and stabilizer groups.

Remark 1. *For $T = \mathrm{Sym}_n(\mathbb{R})$ the Delaunay tessellation of a form in a full-dimensional iso-Delaunay domain is a triangulation. For a general T-space this is no longer true: the Delaunay polytopes of a generic form of T need not be simplices. What characterizes the forms of a full-dimensional domain is that no Delaunay polytope imposes a nonzero linear equality on T; see* (4) *below and Section 2.4.*

2.2. **Linear inequalities from one Delaunay wall.** The fundamental local computation is the construction of the wall inequality associated with a Delaunay polytope P and a neighbouring Delaunay polytope P'. Let $v_1, \dots, v_s \in \mathbb{Z}^n$ be the vertices of P, and choose an affine basis $v_{i_0}, \dots, v_{i_n}$ of P. For a test vertex w, write w as an affine combination of this basis, that is,

$$\sum_{k=0}^{n} \lambda_k = 1, \qquad w = \sum_{k=0}^{n} \lambda_k v_{i_k}. \tag{3}$$

Here "affine basis" means only that the $n+1$ selected vertices are affinely independent, so the coefficients λ_k are real and uniquely determined. This differs from the terminology of Deza and Laurent [7], where an affine basis of a lattice polytope is required to give integral affine coefficients for every vertex of the polytope. That stronger condition cannot be imposed here: there exist Delaunay polytopes having no affine basis in the integral sense [15]. Then the Delaunay condition that w is

outside or on the circumsphere of the basis is the linear condition

$$\sum_{u=1}^{m} q_u \left(\ell_u(w) - \sum_{k=0}^{n} \lambda_k \ell_u(v_{i_k}) \right) \geq 0. \tag{4}$$

When w is itself a vertex of P the left-hand side must vanish, which is a linear equality on T; this linear form is nonzero exactly when P cannot remain a Delaunay polytope on an open subset of T.

For an actual facet adjacency P, P', any vertex of $P' \setminus P$ gives the same inequality up to a positive scalar factor. The scalar is canonicalized by taking the unique primitive integral vector of the ray, which is possible since the coefficients of (4) are rational. All Delaunay-adjacency records producing that same wall are grouped together.

For a Delaunay tessellation D, one loops over all orbits of Delaunay polytopes and over all their facet adjacencies. This gives

$$a_1(q) \geq 0, \ldots, a_N(q) \geq 0, \tag{5}$$

the full half-space description of the closure of the iso-Delaunay domain. Redundant inequalities are then removed, leaving the irredundant wall description used for adjacency. This redundancy removal is carried out by Clarkson's output-sensitive redundancy algorithm [6]. The implementation is the one of [10]: it is built on an exact fraction-free simplex method, whose first phase is run in floating-point arithmetic and its terminal basis then certified, or repaired, in exact arithmetic, and it warm-starts the successive linear programs of Clarkson's algorithm. The same exact simplex method solves the linear programs of Algorithm 2.

Data: A Delaunay tessellation D represented by orbit representatives of Delaunay polytopes and their facet adjacencies; a basis $B_1, \ldots, B_m$ of $T_{\mathbb{Z}}$
Result: The irredundant wall inequalities of the iso-Delaunay cone of D
$\mathcal{A} \leftarrow \emptyset$.
for *each orbit representative P of a Delaunay polytope of D* **do**
  choose an affine basis $v_{i_0}, \ldots, v_{i_n}$ of P.
  for *each facet-neighbour P' of P* **do**
    choose a vertex $w \in P' \setminus P$.
    compute $\lambda_0, \ldots, \lambda_n$ from (3).
    $a(q) \leftarrow$ the linear form occurring in (4).
    replace a by its primitive integral positive multiple.
    append to a the adjacency record (P, P').
    $\mathcal{A} \leftarrow \mathcal{A} \cup \{a\}$, merging equal canonical inequalities and their adjacency records.
  end
end
remove redundant inequalities from $\mathcal{A}$ by Clarkson's algorithm.
return $\mathcal{A}$

Algorithm 1: Wall inequalities of an iso-Delaunay cone

2.3. **A canonical interior point of a polyhedral cone.** The enumeration needs a representative point in the interior of each iso-Delaunay domain. The construction of this point is not specific to iso-Delaunay domains. It applies to any pointed rational polyhedral cone and gives a canonical representative whenever such cones are classified up to the action of an arithmetic group, for instance the perfect domains of Voronoi's algorithm [24, 21] as well as the iso-Delaunay domains considered here. We therefore state it in this generality, and return to iso-Delaunay domains at the end of this section.

Let

$$C = \{q \in \mathbb{R}^m : a_i(q) \geq 0,\ 1 \leq i \leq N\} \tag{6}$$

be a full-dimensional pointed polyhedral cone in $\mathbb{R}^m$ with rational facets. We take for $a_1, \ldots, a_N$ the facet-defining inequalities of C, each normalized to be primitive integral: $a_i(q) = \langle \alpha_i, q \rangle$ with $\alpha_i \in \mathbb{Z}^m$ having coprime coordinates. This list is determined by C up to order. The goal is to choose a point $p(C)$ in the interior of C such that

(i) $p(C)$ depends only on C, and not on the order of the inequalities;
(ii) $p(A\,C) = A\,p(C)$ for every $A \in \mathrm{GL}_m(\mathbb{Z})$.

In particular $p(C)$ is fixed by the stabilizer of C in $\mathrm{GL}_m(\mathbb{Z})$. A generic linear program would find an interior point, but the result would depend on accidental choices in the linear program.

The first linear program. The first step fixes the scale of the cone by working on the affine section

$$S = \{q \in \mathbb{R}^m : a_i(q) \geq 1 \text{ for all } i\}. \tag{7}$$

On this section minimize the symmetric objective

$$\sum_{i=1}^{N} (a_i(q) - 1), \tag{8}$$

which is the sum of all inequalities and is therefore independent of their order. The linear program is bounded and its optimal face

$$F = \left\{ q \in S : \sum_i (a_i(q) - 1) = \mu \right\}, \tag{9}$$

where $\mu \geq 0$ is the optimal value, is a polytope. Indeed, the linear form $\sum_i a_i$ is positive on $C \setminus \{0\}$: if it vanishes at $q \in C$ then every $a_i(q)$ vanishes, and $q = 0$ since C is pointed. If F is a single point, that point is returned. It lies in the interior of C, since $a_i(q) \geq 1 > 0$ for every i.

The degenerate case. The recursion is cleanest in homogeneous coordinates, and this is how it is implemented. Put $\hat{q} = (t, q) \in \mathbb{R}^{1+m}$ and consider the cone

$$K = \{(t, q) : a_i(q) - t \geq 0,\ 1 \leq i \leq N\}, \tag{10}$$

whose section $t = 1$ is S. The optimal value defines the hyperplane

$$H = \left\{ (t, q) : \sum_{i=1}^{N} a_i(q) - (N + \mu)t = 0 \right\}, \tag{11}$$

whose section $t = 1$ is the affine span of the optimal objective level. Inside the linear space H, some of the inequalities $a_i(q) - t \geq 0$ are equalities on all of $K \cap H$.

Let $W \subseteq H$ be the linear space they define and let I be the set of the remaining inequalities, those that are not identically zero on W. Then $K \cap W$ is a full-dimensional cone of W described by the inequalities in I, and its section $t = 1$ is F.

Lemma 1. *If F is not a single point, then $\mu > 0$ and $t > 0$ on $(K \cap W) \setminus \{0\}$.*

Proof. If $\mu = 0$, the optimum is attained exactly at the points with $a_i(q) = 1$ for all i. Two distinct such points differ by a nonzero d with $a_i(d) = 0$ for all i, which contradicts the pointedness of C. Hence $\mu > 0$. For $(t, q) \in K \cap H$, summing the inequalities $a_i(q) \geq t$ gives $(N + \mu)t = \sum_i a_i(q) \geq Nt$, that is $\mu t \geq 0$, so $t \geq 0$. If $t = 0$, then every $a_i(q) \geq 0$ and $\sum_i a_i(q) = 0$, so $q = 0$ as above. □

The construction is now applied recursively to the cone $K \cap W$ of W, in coordinates of W. By Lemma 1 this cone is pointed and the point (t_0, q_0) returned by the recursion has $t_0 > 0$; the algorithm returns q_0/t_0. This is a point of the relative interior of F, hence an interior point of C. The recursion terminates since $\dim W < \dim H < 1 + m$. Indeed the objective is not constant on S, which contains $q + \lambda q'$ for $q \in S$, q' in the interior of C and $\lambda \geq 0$; hence F is a proper face of S, at least one inequality $a_i(q) \geq 1$ is tight on F, and the corresponding inequality $a_i(q) - t \geq 0$ is an equality on $K \cap H$.

Equivariance. This construction uses only linear programming and linear algebra; no full dual-description computation is needed. Permuting the inequalities does not change the point. The recursion only uses the restrictions of the forms $a_i(q) - t$ to W, which do not depend on the basis chosen for W. Let $A \in \mathrm{GL}_m(\mathbb{Z})$ act on coordinates by $q \mapsto Aq$. The facet inequalities of $A\,C$ are the forms $a_i \circ A^{-1}$, and since A^{-1} is an integral unimodular matrix these are again primitive integral. Hence the normalized inequalities of $A\,C$ are exactly the transforms of those of C, each step above commutes with A, and $p(A\,C) = A\,p(C)$. Equivariance does not hold for an arbitrary linear change of coordinates, nor for an individual rescaling of the inequalities, because the primitive normalization depends on the lattice $\mathbb{Z}^m$. The lattice must therefore be fixed before the point is computed.

Data: A full-dimensional pointed cone $C = \{q \in \mathbb{R}^m : a_i(q) \geq 0\}$, given by its facet inequalities a_i, each primitive integral
Result: A point $p(C)$ in the interior of C, equivariant under $\mathrm{GL}_m(\mathbb{Z})$
Solve the linear program minimizing $\sum_i (a_i(q) - 1)$ subject to $a_i(q) \geq 1$; let μ be the optimal value and q^* a solution.
if *the optimal face is the single point* q^* **then**
  return q^*
end
$K \leftarrow \{(t, q) : a_i(q) - t \geq 0\}$, $H \leftarrow \{(t, q) : \sum_i a_i(q) - (N + \mu)t = 0\}$.
$W \leftarrow$ the subspace of H cut out by the inequalities that are equalities on $K \cap H$.
Express the inequalities that are not identically zero on W in a basis of W.
$(t_0, q_0) \leftarrow$ result of Algorithm 2 applied to this cone of W, mapped back to $\mathbb{R}^{1+m}$.
return q_0/t_0

Algorithm 2: Geometrically unique interior point

The point of this canonical choice is the following statement, which reduces the equivalence of cones to the equivalence of points.

Lemma 2. *Let $\Gamma \subseteq \mathrm{GL}_m(\mathbb{Z})$ be a group and let $\mathcal{F}$ be a Γ-invariant family of full-dimensional pointed rational polyhedral cones in $\mathbb{R}^m$ with pairwise disjoint interiors. For $C, C' \in \mathcal{F}$, we have $C' = A\,C$ for some $A \in \Gamma$ if and only if $p(C') = A\,p(C)$ for some $A \in \Gamma$.*

Proof. If $C' = A\,C$, then $p(C') = A\,p(C)$ by equivariance. Conversely, if $p(C') = A\,p(C)$, then this point lies in the interior of C' and in the interior of $A\,C \in \mathcal{F}$. Since distinct cones of $\mathcal{F}$ have disjoint interiors, $C' = A\,C$. □

Application to iso-Delaunay domains. In the coordinates q of the previous sections the lattice $\mathbb{Z}^m$ is $T_{\mathbb{Z}}$, and $\mathrm{GL}(T_{\mathbb{Z}})$ is a subgroup of $\mathrm{GL}_m(\mathbb{Z})$. The closure C of a full-dimensional iso-Delaunay domain is pointed, since it is contained in the cone of positive semidefinite matrices, and its facet inequalities are the primitive integral ones produced by Algorithm 1. We write Q_C for the Gram matrix of the point $p(C)$ and call it the interior form of the domain. The full-dimensional iso-Delaunay domains have pairwise disjoint interiors and are permuted by $\mathrm{GL}(T_{\mathbb{Z}})$. Lemma 2 with $\Gamma = \mathrm{GL}(T_{\mathbb{Z}})$ therefore shows that two such domains are equivalent if and only if their interior forms are; this is what makes the equivalence test of Section 2.6 correct.

2.4. **Construction of an initial form.** It remains to construct a form lying in a full-dimensional iso-Delaunay domain. This is done iteratively. Fix a positive-definite matrix $Q_{\mathrm{ref}} \in T$. Choose integral coefficients c_u, form

$$Q = \sum_{u=1}^{m} c_u B_u, \tag{12}$$

and add sufficiently many copies of Q_{ref} to make Q positive definite. If an arithmetic automorphism of Q does not fix the T-space pointwise, discard Q and sample again. The automorphisms that fix T pointwise are automorphisms of every form of T; a form with more symmetry is special in T, and discarding it is a cheap way to avoid forms lying on walls.

The random generation starts with simple forms: the coefficients c_u are sampled uniformly in $\{-N, \dots, N\}$, starting from $N = 2$. If the k-th candidate fails the tests below, N is increased by k, so that $N = 2 + k(k+1)/2$ after k failures. The precise rate of growth is not part of the mathematical algorithm; its purpose is to explore low-complexity forms first without restricting the search to a fixed finite set.

For a form passing this symmetry test, enumerate its Delaunay tessellation. This enumeration may be stopped as soon as a Delaunay polytope imposes a nonzero linear equality on T. Such an equality means that Q lies on a proper iso-Delaunay wall rather than in a full-dimensional domain.

Once a tessellation D_0 passes all tests, its wall inequalities are computed and Algorithm 2 supplies the distinguished interior form Q_0. The iterative construction is summarized in Algorithm 3.

A prescribed common form. The enumeration may be restricted to the *star* of a prescribed positive-definite form $G = Q_{\mathrm{common}} \in T$: the set of full-dimensional iso-Delaunay domains whose closure contains G. This is used when G lies on walls,

for instance when G has a non-generic Delaunay tessellation. The seed must then be a domain of the star. Since the domains are cones, the form $P + sG$ with P a generic form tends to the ray of G as $s \to \infty$, and lies in a domain of the star for s large enough. The algorithm therefore samples P as above, tests $Q = P + 2^k G$ at the k-th attempt, and accepts a full-dimensional domain only if G belongs to its closure, that is, if $a(G) \geq 0$ for all its wall inequalities a. This test is made once for the whole domain after its Delaunay tessellation has been computed.

Data: The saturated T-space with basis $B_1, \ldots, B_m$, a positive-definite form $Q_{\text{ref}} \in T$, and optionally a positive-definite form $G = Q_{\text{common}} \in T$
Result: A Delaunay tessellation D_0 of a full-dimensional iso-Delaunay domain, whose closure contains G if G is given, and its distinguished interior form Q_0

$k \leftarrow 0$, $N \leftarrow 2$.
while *no initial domain has been found* **do**
  Choose random integral coefficients $c_1, \ldots, c_m$ in $\{-N, \ldots, N\}$.
  $Q \leftarrow \sum_{u=1}^{m} c_u B_u$.
  while Q *is not positive definite* **do**
    $Q \leftarrow Q + Q_{\text{ref}}$.
  end
  if *some arithmetic automorphism of* Q *does not fix* T *pointwise* **then**
    $k \leftarrow k + 1$, $N \leftarrow N + k$, and continue with another sampled form.
  end
  if G *is prescribed* **then**
    $Q \leftarrow Q + 2^k G$.
  end
  Enumerate the Delaunay tessellation D of Q, stopping if a Delaunay polytope imposes a nonzero equality on T.
  if *the enumeration finishes without stopping* **then**
    $\mathcal{A}_0 \leftarrow$ wall inequalities of D by Algorithm 1.
    if G *is not prescribed, or* $a(G) \geq 0$ *for all* $a \in \mathcal{A}_0$ **then**
      $Q_0 \leftarrow$ result of Algorithm 2 applied to $\mathcal{A}_0$.
      return (D, Q_0)
    end
  end
  $k \leftarrow k + 1$, $N \leftarrow N + k$.
end

Algorithm 3: Iterative construction of an initial iso-Delaunay domain

2.5. **Adjacent domains and flips.** The adjacency graph of iso-Delaunay domains is obtained by crossing irredundant walls of the current cone. For each orbit representative of an irredundant inequality, the algorithm first computes a point in the relative interior of the corresponding facet. If this point does not represent a positive-definite matrix, the wall belongs to the boundary of the positive definite cone and no adjacent iso-Delaunay domain is produced. Otherwise the wall is crossed by a T-flip. Testing a single point is enough by the following remark.

Remark 2. *Let X be a convex subset of the cone of positive semidefinite matrices. All the points of the relative interior of X lie in the relative interior of the same face of that cone, and the faces of the positive semidefinite cone are determined by the kernel. Hence all the points of* $\operatorname{relint} X$ *have the same rank. Applied to a facet of an iso-Delaunay cone, either every point of its relative interior is positive definite, or the facet lies in the boundary of the positive semidefinite cone.*

When a common form G is prescribed, a wall is crossed only if G lies on it, that is, if its inequality vanishes at G. Crossing a wall not containing G would leave the star of G. Because G is positive definite, the domains of the star form a complete fan around the ray of G modulo the line $\mathbb{R}G$, since the iso-Delaunay domains form a locally finite decomposition of the positive definite cone [16], and any two of them are therefore connected by a sequence of crossings of walls containing G.

The flip uses the adjacency records stored with the wall inequality to identify all Delaunay polytopes that meet at the wall. These polytopes form repartitioning polytopes. The repartitioning construction lifts vertices by the height x^tQx for an interior form Q of the old domain. Its facets split into three classes:

- lower facets, which are the old Delaunay polytopes;
- higher facets, which become the new Delaunay polytopes after crossing the wall;
- barrel facets, which transmit adjacencies between neighbouring repartitioning polytopes.

After the higher facets have been identified modulo the local symmetry group, one reconstructs the whole finite orbit description of the new Delaunay tessellation and recomputes its facet adjacencies. All the flips across the walls of one domain start from the same tessellation D, so the data depending only on D, such as the stabilizers of its Delaunay polytopes, are computed once and shared by these flips.

2.6. **Equivalence and graph traversal.** Enumeration is a graph traversal in which each visited vertex is an iso-Delaunay domain and each edge is a positive-definite wall crossing. The data required by this traversal are:

- construction of one initial iso-Delaunay domain;
- computation of all adjacent domains by the wall-and-flip procedure above;
- computation of a fast invariant, for instance from an invariant short-vector family of the interior Gram matrix;
- an exact equivalence test between two domains, obtained by finding a T-space-preserving integral change of basis sending one interior Gram matrix and short-vector family to the other. Because the matrix lattice $T_{\mathbb{Z}}$ is saturated, the induced transformation matrix in the chosen T-coordinates is integral. By Lemma 2, comparing the interior Gram matrices produced by Algorithm 2 decides the equivalence of the domains.

When a common form G is prescribed, the arithmetic group is replaced by the stabilizer of G in it, so that the equivalences preserve the star of G. The output of the traversal is therefore one representative for each iso-Delaunay domain modulo the arithmetic group preserving the T-space.

2.7. **Enumeration algorithm.** The top-dimensional enumeration is obtained by putting the preceding subroutines together. Its structure is the usual adjacency

Data: An iso-Delaunay domain C, represented by its Delaunay tessellation D, an interior form Q, and the wall inequalities of C; a set $\mathcal{K}$ of walls of C whose adjacent domain is already known; optionally a common form G

Result: The list of positive-definite iso-Delaunay domains adjacent to C, one for each orbit of walls of C under its stabilizer that meets no wall of $\mathcal{K}$

$\mathcal{N} \leftarrow \emptyset$.
Compute the stabilizer of C in the arithmetic group preserving T.
Compute orbit representatives of the irredundant walls of C, taking the walls of $\mathcal{K}$ as the representatives of their orbits.
for *each wall representative* $F \notin \mathcal{K}$, *with inequality* a_F **do**
  compute a point q_F in the relative interior of F.
  if $Q(q_F)$ *is not positive definite* **then**
    continue to the next wall.
  end
  if G *is prescribed and* $a_F(G) \neq 0$ **then**
    continue to the next wall.
  end
  collect all Delaunay adjacencies producing the wall F.
  construct the associated repartitioning polytopes.
  replace lower facets by higher facets, keeping barrel facets for adjacency matching.
  reconstruct the finite orbit description of the flipped Delaunay tessellation D'.
  compute an interior form Q' of the cone of D' by Algorithm 2.
  $\mathcal{N} \leftarrow \mathcal{N} \cup \{(D', Q')\}$.
end
return $\mathcal{N}$

Algorithm 4: Adjacent iso-Delaunay domains

enumeration: maintain a queue of domains still to be processed, compute all neighbours of one domain, and insert only those neighbours that are not equivalent to previously found domains.

The adjacency relation is symmetric, and this saves flips. Suppose that crossing the wall F of C, with inequality a_F, gives a domain C', and that $C' = g\,C''$ for a domain C'' of the queue and an element g of the arithmetic group. Then C' has the wall F, with inequality $-a_F$, and crossing it leads back to C. Hence the wall $g^{-1}F$ of C'' leads to a domain equivalent to C, which need not be computed when C'' is processed. Such walls are recorded with the domains of the queue and passed to Algorithm 4. In this way each orbit of walls is crossed from one side only, which roughly halves the number of flips.

3. Test cases: Hermitian forms over imaginary quadratic rings

The enumeration of Section 2 applies to any T-space, and the spaces of Hermitian forms over the ring of integers of an imaginary quadratic field give test cases in which the arithmetic group is not $\mathrm{GL}_n(\mathbb{Z})$. These spaces and their perfect forms are studied in [13], whose model we use.

Data: A T-space with basis $B_1, \dots, B_m$, an arithmetic group preserving T, and optionally a common Gram matrix Q_{common}
Result: A set $\mathcal{R}$ of inequivalent full-dimensional iso-Delaunay domains in T, restricted to the star of Q_{common} if it is given
$(D_0, Q_0) \leftarrow$ result of Algorithm 3.
$\mathcal{T} \leftarrow \{(D_0, Q_0)\}$, $\mathcal{R} \leftarrow \emptyset$, $\mathcal{K}(D_0, Q_0) \leftarrow \emptyset$.
while $\mathcal{T} \neq \emptyset$ **do**
  choose $C = (D, Q) \in \mathcal{T}$.
  $\mathcal{T} \leftarrow \mathcal{T} \setminus \{C\}$.
  $\mathcal{R} \leftarrow \mathcal{R} \cup \{C\}$.
  $\mathcal{N} \leftarrow$ adjacent domains of C by Algorithm 4, with the known walls $\mathcal{K}(C)$.
  for $C' \in \mathcal{N}$, *obtained by crossing the wall* F *of* C **do**
    if $C' = g\,C''$ *for some* $C'' \in \mathcal{R} \cup \mathcal{T}$ **then**
      if $C'' \in \mathcal{T}$ **then**
        $\mathcal{K}(C'') \leftarrow \mathcal{K}(C'') \cup \{g^{-1}F\}$.
      end
    else
      $\mathcal{T} \leftarrow \mathcal{T} \cup \{C'\}$, $\mathcal{K}(C') \leftarrow \{F\}$.
    end
  end
end
return $\mathcal{R}$

Algorithm 5: Enumeration of full-dimensional iso-Delaunay domains

The T-space. Let $R = \mathbb{Z}[\alpha]$ be the ring of integers of an imaginary quadratic field, with $\alpha^2 - s\alpha + p = 0$, $s \in \{0, 1\}$; the Gaussian integers $\mathbb{Z}[i]$ have $(s, p) = (0, 1)$ and the Eisenstein integers $(s, p) = (1, 1)$. We identify R^n with $\mathbb{Z}^{2n}$ by writing $x = a + \alpha b$ with $a, b \in \mathbb{Z}^n$. A Hermitian form H on R^n defines the real quadratic form $(a, b) \mapsto H(x, x)$ on $\mathbb{Z}^{2n}$, and $T \subset \mathrm{Sym}_{2n}(\mathbb{R})$ is the space of these forms, of dimension n^2; its explicit description as a space of symmetric $2n \times 2n$ matrices is given in [13, Section 6.1]. Let $M_\alpha \in \mathrm{GL}_{2n}(\mathbb{Z})$ be the matrix of the multiplication by α. Since $H(\alpha x, \alpha x) = p\,H(x, x)$, every $A \in T$ satisfies $M_\alpha^t A M_\alpha = pA$.
The groups. The group $\mathrm{GL}_n(R)$ is the group of the matrices of $\mathrm{GL}_{2n}(\mathbb{Z})$ that commute with M_α; it preserves T. The Galois conjugation $\sigma\colon a + \alpha b \mapsto a + \bar{\alpha} b$ is also a matrix of $\mathrm{GL}_{2n}(\mathbb{Z})$ preserving T, which normalizes $\mathrm{GL}_n(R)$, and we also consider the group $K_n(R) = \mathrm{GL}_n(R) \cup \mathrm{GL}_n(R)\,\sigma$. The stabilizers and the equivalences of positive definite forms in $\mathrm{GL}_n(R)$ are computed by the algorithm of Plesken and Souvignier [20] applied to the pair of forms A and AM_α: an element of $\mathrm{GL}_{2n}(\mathbb{Z})$ preserving both commutes with M_α. The elements of the coset $\mathrm{GL}_n(R)\,\sigma$ do not commute with M_α and are reduced to $\mathrm{GL}_n(R)$: the element $g\sigma$ maps A_1 to A_2 if and only if g maps $\sigma A_1 \sigma^t$ to A_2. The group $K_n(R)$ can also be obtained without this reduction, as the group of all the elements of $\mathrm{GL}_{2n}(\mathbb{Z})$ preserving T, and the two computations have to agree.

Table 1 gives the numbers of inequivalent full-dimensional iso-Delaunay domains for $n = 2$ and the first ten discriminants, and for $n = 3$ over the Gaussian and the Eisenstein integers, with the running times of the enumeration. For $n = 3$ over $\mathbb{Z}[i]$,

the computation of $K_3(\mathbb{Z}[i])$ as the group of all the elements preserving T gives the same 103 domains.

TABLE 1. Number of inequivalent full-dimensional iso-Delaunay domains of Hermitian forms over the ring of integers of $\mathbb{Q}(\sqrt{d})$, and the wall-clock time of their enumeration in seconds, on a single core of an Apple M4 Max.

n	d	$\mathrm{GL}_n(R)$	time (s)	$K_n(R)$	time (s)
2	−3	1	0.06	1	0.02
2	−4	1	0.03	1	0.03
2	−7	16	0.06	11	0.05
2	−8	5	0.04	5	0.04
2	−11	58	0.16	35	0.10
2	−15	127	0.32	74	0.20
2	−19	198	0.52	112	0.31
2	−20	61	0.18	43	0.14
2	−23	343	0.90	190	0.53
2	−24	86	0.26	61	0.22
3	−3	23413	1061	11832	401
3	−4	206	8.9	103	4.9

4. Finding iso-Delaunay domains with all rays of full rank

The following search is more targeted than full enumeration. Its purpose is to find, in a fixed dimension n, a top-dimensional iso-Delaunay domain whose every extreme ray gives a full-rank Gram matrix. The search is not dimension-specific.

4.1. **The objective function.** Let $C \subset T$ be the closure of an iso-Delaunay domain with irredundant inequality description. A dual-description computation returns its extreme rays

$$r^{(1)}, \dots, r^{(s)} \in \mathbb{R}^m. \tag{13}$$

Each ray is interpreted as a symmetric matrix

$$R_i = \sum_{u=1}^{m} r_u^{(i)} B_u. \tag{14}$$

The desired property is

$$\operatorname{rank}(R_i) = n \qquad \text{for every extreme ray } i. \tag{15}$$

Since C is contained in the cone of positive semidefinite matrices, each R_i is positive semidefinite, and the property says that every R_i is positive definite. As C is the conical hull of its extreme rays, it is then equivalent to $C \setminus \{0\}$ consisting of positive definite forms, and it implies, by Remark 2, that every wall of C is crossed by a flip into another iso-Delaunay domain. The search uses the integer objective

$$\Phi(C) = \sum_{i=1}^{s} (n - \operatorname{rank}(R_i)). \tag{16}$$

Thus $\Phi(C) = 0$ exactly when all extreme rays have full rank. For each visited domain, the inequalities are recomputed, redundancy is removed by Clarkson's algorithm, the extreme rays are computed, and the rank defects are summed. This makes the objective independent of any redundant wall description used during adjacency generation. The dual descriptions are the dominant cost of the search; the implementation chooses among several dual description methods adaptively during the run.

Remark 3. *For $T = \mathrm{Sym}_n(\mathbb{R})$ with $2 \leq n \leq 5$, no domain has $\Phi = 0$. For $n = 5$, the complete classification of the L-types* [14] *shows that every secondary cone C decomposes as $C = F(C) + \sum_{i=1}^{h} \mathbb{R}_+ v_i v_i^t$, where the fundamental face $F(C)$ is the smallest face of C containing all its extreme rays of rank greater than 1, and $\dim C = \dim F(C) + h$. The fundamental faces that occur have dimension at most 5, so each of the 222 full-dimensional domains, of dimension 15, has at least 10 extreme rays of rank 1. For $n = 4$ there are three full-dimensional domains* [24]*, with 10, 9 and 9 extreme rays of rank 1, and for $n = 2, 3$ the only full-dimensional domain, Voronoi's principal domain, has all its extreme rays of rank 1.*

Data: An iso-Delaunay domain C with Delaunay tessellation D
Result: The rank-defect score $\Phi(C)$
Compute the wall inequalities of D by Algorithm 1.
Remove redundant inequalities by Clarkson's algorithm.
Compute the extreme rays $r^{(1)}, \ldots, r^{(s)}$ of the resulting cone.
$\Phi \leftarrow 0$.
for $i = 1, \ldots, s$ **do**
  $R_i \leftarrow \sum_{u=1}^{m} r_u^{(i)} B_u$.
  $\Phi \leftarrow \Phi + (n - \mathrm{rank}(R_i))$.
end
return Φ

Algorithm 6: Rank-defect objective

4.2. **Greedy descent with random escapes.** The search starts from a generic iso-Delaunay domain obtained by Algorithm 3. The strategy is the same one used in Frank Vallentin's PhD thesis [23]: move greedily in the graph of adjacent L-types when the objective improves, and use random walks to leave local minima. In that thesis this strategy was used to find record lattice coverings in dimension 6.

The neighbours of a domain are taken from Algorithm 4, that is, one neighbour for each orbit of flippable walls under the stabilizer of the domain. This is enough for the greedy step, because Φ is invariant under the arithmetic group and two walls in the same orbit lead to equivalent neighbours. For the random walk it means that each step chooses uniformly among the orbits of flippable walls, which is not the same distribution as choosing uniformly among the walls themselves.

The output condition is exactly $\Phi(C) = 0$. In this case every extreme ray of the iso-Delaunay domain has full rank.

4.3. **Search parameters.** The search is controlled by the following parameters:

- the length of the random escape walk after a local minimum;
- the cap on the number of outer iterations.

Data: An n-dimensional T-space, a random-walk length N_{walk}, and an iteration bound N_{iter}
Result: A domain C with $\Phi(C) = 0$, if one is found within the search budget
$C \leftarrow$ result of Algorithm 3.
$\phi \leftarrow \Phi(C)$, computed by Algorithm 6.
for $t = 1, \dots, N_{\mathrm{iter}}$ **do**
  if $\phi = 0$ **then**
    return C
  end
  $\mathcal{N} \leftarrow$ adjacent domains of C by Algorithm 4.
  compute $\Phi(C')$ for all $C' \in \mathcal{N}$.
  $\phi_{\min} \leftarrow \min\{\Phi(C') : C' \in \mathcal{N}\}$.
  if $\phi_{\min} < \phi$ **then**
    choose uniformly a domain $C' \in \mathcal{N}$ with $\Phi(C') = \phi_{\min}$.
    $C \leftarrow C'$, $\phi \leftarrow \phi_{\min}$.
  else
    for $j = 1, \dots, N_{\mathrm{walk}}$ **do**
      choose uniformly an orbit of flippable walls of C and replace C by the domain across a wall of that orbit.
    end
    recompute $\phi = \Phi(C)$.
  end
end
return fail

Algorithm 7: Search for an iso-Delaunay domain with full-rank rays

The iteration cap is part of the mathematical design of the search, not only a safety guard. In some dimensions there is no domain with $\Phi = 0$, as Remark 3 shows, so an unconditional search for a zero would not terminate. The search exploits the adjacency graph of iso-Delaunay domains until such a domain is found or the chosen computational budget is exhausted.

5. Iso-Delaunay domains in dimension 6 with full-rank rays

We ran Algorithm 7 for $T = \mathrm{Sym}_6(\mathbb{R})$, the classical case of Voronoi's second reduction, where the iso-Delaunay domains are the L-type domains. Here $m = 21$, and the basis $B_1, \dots, B_{21}$ consists of the matrices E_{ii} and $E_{ij} + E_{ji}$ for $i < j$, in the order $(1,1), (1,2), \dots, (1,6), (2,2), \dots, (6,6)$, so that q_{ij} is the (i,j) entry of $Q(q)$. Each full-dimensional L-type domain determines a combinatorial type of primitive parallelohedra [24]. In dimension 6, Baburin and Engel found 567 613 632 of these types without completing the enumeration [3], so the complete list of L-type domains is not known. Our search does not attempt it: it looks only for the domains all of whose extreme rays are positive definite. The search was run with $N_{\mathrm{walk}} = 50$ and $N_{\mathrm{iter}} = 200$.

Theorem 2. *There exist* 21*-dimensional L-type domains of positive definite quadratic forms in six variables all of whose extreme rays are positive definite forms. The search found* 1472 *of them, pairwise inequivalent under* $\mathrm{GL}_6(\mathbb{Z})$*. Their number of facets ranges from* 24 *to* 62*, and their number of extreme rays from* 47 *to* 210010*.*

The data. The search saved 1472 Gram matrices. For each of them we recomputed, with the programs of [10], its Delaunay tessellation, the iso-Delaunay domain containing it, the irredundant facets of the domain and its extreme rays. All 1472 domains have $\Phi = 0$.

This recomputation has to take the symmetries of the saved form into account. The Delaunay tessellation of a form Q is stored as one Delaunay polytope per orbit of the stabilizer of Q in $\mathrm{GL}_6(\mathbb{Z})$, and the walls of the other polytopes are the images of the walls of these representatives under the stabilizer. The element -1 acts trivially on T, but a larger stabilizer does not, so the list of walls has to be closed under it before the redundancy is removed; otherwise the cone obtained is not the iso-Delaunay domain. For 1457 of the saved forms the stabilizer is $\{\pm 1\}$, and for the other 15 it has order 4. The enumeration of Section 2 is not concerned: it computes one Delaunay tessellation, for a form without accidental symmetry, and obtains the other tessellations by flips, so that its lists of Delaunay polytopes are never reduced by the stabilizer of a later domain.

The 1472 domains are pairwise inequivalent under $\mathrm{GL}_6(\mathbb{Z})$. Since all extreme rays of a domain C are positive definite, the primitive integral extreme rays are known, and their sum R_C is a point in the interior of C. The point R_C is equivariant under $\mathrm{GL}_6(\mathbb{Z})$, and the proof of Lemma 2 uses only these two properties of $p(C)$, so two domains are equivalent if and only if their points R_C are. The 1472 forms R_C have pairwise distinct canonical forms under $\mathrm{GL}_6(\mathbb{Z})$, computed with [10].

The smallest example. The domain with the fewest facets and rays has 24 facets and 47 extreme rays. We give it in the basis of $\mathbb{Z}^6$ in which the Gram matrix saved by the search is Minkowski reduced, computed with [10]. In this basis the saved Gram matrix, which lies in the interior of the domain, is

$$Q_0 = \begin{pmatrix} 130 & 61 & -44 & -36 & -50 & -18 \\ 61 & 131 & -36 & 26 & -62 & 17 \\ -44 & -36 & 153 & -55 & -33 & 67 \\ -36 & 26 & -55 & 160 & 33 & 34 \\ -50 & -62 & -33 & 33 & 176 & -83 \\ -18 & 17 & 67 & 34 & -83 & 183 \end{pmatrix}, \tag{17}$$

the facet inequalities have coefficients in $\{-2, \ldots, 2\}$, and the extreme rays have coordinates of absolute value at most 26. No integral form in the interior of the domain has smaller coefficients in this basis: the linear program minimizing the largest absolute value of a coefficient, subject to $a(q) \geq 1$ for all facets a, already has the optimal value 183. All 47 extreme rays are positive definite forms, with determinants ranging from 24 to 21208743 for the primitive integral representatives. The facets and the extreme rays are listed in Appendix A.

Data availability

The 1472 domains of Theorem 2 are available on Zenodo [12]: for each of them a Gram matrix in its interior and its facets, from which the extreme rays can be recomputed.

Acknowledgments

The author thanks Barak Weiss for suggesting the question of the existence of iso-Delaunay domains all of whose extreme rays have full rank.

Tool and computational resource disclosure

This section follows the recommendation of the Leiden Declaration on Artificial Intelligence and Mathematics [1].

Mathematical software. The algorithms of Sections 2 and 4 are implemented in the C++ programs of [10], with no external library for linear programming or polyhedral computations: the exact simplex method and Clarkson's redundancy removal are implemented there. The dual description is computed in [10] by using the classical methods of reverse search [2], Normaliz [5], double description [19], beneath-and-beyond [17] and primal-dual [4], as well as the recursive adjacency decomposition method [11]. The data of Section 5 were processed with Python scripts calling the programs of [10], including its canonical form of positive definite forms under $\mathrm{GL}_n(\mathbb{Z})$.

Large language models. The AI assistant Claude (Claude Opus 5.5, by Anthropic), through the Claude Code interface, assisted with this version of the manuscript in November 2026: it compared the description of the algorithms with their implementation in [10] and drafted the resulting revisions, drafted Section 5 and Appendix A, and recomputed the data of Section 5, which led to the correction of the analysis program described there. The search itself was performed by the programs of [10]. The author has checked the mathematical content, the proofs and the computations, and takes full responsibility for the correctness of the paper and of its citations.

Appendix A. The smallest domain of Theorem 2

Facets. The 24 irredundant wall inequalities $a(q) \geq 0$, one per row, in the coordinates $q_{11}, q_{12}, \ldots, q_{16}, q_{22}, \ldots, q_{66}$ of Section 5.

0	0	0	-1	0	1	0	0	0	0	-1	0	0	-1	0	0	-1	0	0	0	0
0	0	0	-1	1	1	0	0	1	-1	-1	0	-1	0	0	-1	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1	0
0	1	1	0	1	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
0	0	1	1	0	-1	0	-1	-1	0	1	0	1	0	0	1	-1	-1	0	0	0
0	0	0	-1	1	1	0	0	0	0	-1	0	0	1	0	0	0	0	0	-1	0
0	0	1	0	0	-1	0	-1	-1	0	1	0	0	0	0	0	0	0	0	0	0
0	-1	0	0	-1	0	0	-1	0	0	0	0	1	-2	-2	0	-1	-2	0	2	2
0	0	0	0	-1	0	0	0	-1	1	1	0	-1	0	0	0	-1	0	0	0	0
0	0	0	0	0	-1	0	0	0	0	1	0	0	0	0	0	0	-1	0	0	0
0	-1	1	1	-1	-1	1	0	-1	1	0	1	1	0	-1	0	0	0	0	0	0
0	0	-1	0	0	1	0	1	1	-1	-1	0	0	0	0	0	1	0	-1	-1	0
-1	1	-1	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	1	0	-1	0	1	-1	0	0	0	0	0	0	1	0	0	0	0
0	0	0	1	-1	-1	0	-1	-1	1	1	0	1	-1	-1	1	-1	-1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	1	0	-1	-1
0	0	0	0	-1	-1	0	0	0	0	0	0	0	-1	-1	0	-1	0	0	0	0
0	-1	-1	0	-1	1	0	0	0	0	-1	-1	-1	0	1	0	0	1	0	0	0
1	0	1	1	1	0	0	0	0	0	0	0	0	1	1	0	1	1	0	-1	-1
0	1	0	0	1	0	0	1	1	0	0	0	0	1	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	-1	-1
0	0	-1	0	0	1	0	0	-1	1	0	-1	-1	0	1	0	-1	1	1	1	0
0	-1	0	1	-1	0	0	-1	-1	0	0	0	1	0	0	1	-1	-1	0	0	0
-1	1	-1	-1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0

Extreme rays. The 47 extreme rays, as primitive integral vectors in the same coordinates, sorted by increasing determinant of the matrix R_i. Each R_i is positive definite.

2	1	0	-1	-1	0	2	0	0	-1	0	2	-1	-1	1	3	1	0	3	-1	2
2	1	0	-1	-1	0	3	0	0	-2	1	2	-1	-1	1	4	1	1	4	-2	4
3	1	-1	-1	-1	-1	3	-1	1	-1	0	4	-2	-1	2	5	1	1	4	-2	5
4	2	-2	-1	-1	-1	4	-2	1	-1	0	5	-2	-1	2	5	1	1	4	-2	5
4	2	-1	-1	-2	0	4	-1	1	-2	1	4	-1	-1	2	4	1	1	6	-3	6
4	2	-1	-1	-2	0	4	-1	1	-2	1	5	-2	-1	2	5	1	1	6	-3	6
5	3	-1	-2	-3	0	5	-1	0	-3	1	4	-1	-1	2	5	2	1	7	-3	6
5	3	-2	-1	-2	-1	5	-2	1	-2	0	6	-2	-1	3	5	1	1	6	-3	7
4	2	-1	-1	-2	0	5	-1	1	-3	1	5	-2	-1	2	6	1	1	7	-3	6
4	2	-1	-1	-2	0	6	-1	2	-3	2	6	-2	-2	3	7	1	2	9	-5	9
7	4	-2	-2	-4	0	7	-2	1	-4	1	7	-2	-1	3	7	2	1	10	-4	8
7	4	-2	-2	-4	0	8	-2	2	-4	2	8	-2	-2	4	8	2	2	12	-6	11
6	3	-1	-2	-3	0	8	-1	2	-4	2	7	-3	-2	3	10	2	2	11	-5	10
8	4	-2	-2	-4	0	8	-2	2	-4	2	8	-2	-2	4	8	2	2	12	-6	11
8	4	-3	-2	-3	-1	8	-2	1	-4	1	9	-3	-2	4	9	2	2	10	-5	11
8	4	-3	-2	-3	-1	8	-2	1	-4	1	9	-3	-2	4	9	2	2	11	-5	11
8	4	-3	-2	-3	-1	9	-3	2	-4	1	10	-4	-2	4	10	2	2	11	-5	11
9	5	-2	-3	-5	0	10	-2	2	-5	2	9	-3	-2	4	11	3	2	14	-6	12
10	5	-2	-3	-5	0	10	-2	2	-5	2	9	-3	-2	4	11	3	2	14	-6	12
10	4	-4	-3	-3	-2	9	-3	2	-4	1	12	-4	-2	5	12	2	3	11	-5	13
9	4	-3	-2	-3	-1	10	-3	3	-4	2	11	-4	-3	5	12	2	3	13	-7	14
10	4	-4	-3	-3	-2	9	-3	2	-4	1	12	-4	-2	5	12	2	3	12	-6	14
11	5	-4	-3	-4	-2	10	-3	2	-5	1	12	-4	-2	5	12	2	3	13	-6	14
10	5	-3	-2	-4	-1	11	-3	3	-5	2	12	-4	-3	6	12	2	3	15	-8	16
11	5	-4	-3	-4	-2	11	-3	2	-5	1	13	-5	-3	6	13	3	2	15	-7	15
11	5	-4	-3	-4	-2	10	-3	2	-5	1	14	-5	-3	6	14	3	3	15	-7	16
11	5	-4	-3	-4	-2	11	-3	2	-5	1	14	-5	-3	6	14	3	3	15	-7	16
11	5	-3	-3	-4	-1	12	-3	3	-5	2	12	-5	-3	5	15	3	3	15	-7	15
12	6	-3	-3	-5	-1	13	-3	3	-6	2	13	-5	-3	6	15	3	3	17	-8	17
13	6	-4	-3	-5	-1	14	-3	3	-7	3	15	-5	-4	7	16	3	4	19	-10	20
13	6	-4	-3	-5	-1	14	-3	3	-7	3	15	-5	-4	7	16	3	4	20	-10	20
13	6	-4	-3	-5	-1	15	-4	4	-7	3	16	-6	-4	7	17	3	4	20	-10	20
15	7	-4	-4	-6	-1	16	-3	3	-8	3	16	-6	-4	7	19	4	4	21	-10	21
15	6	-5	-4	-5	-2	15	-4	4	-7	3	18	-6	-4	8	19	3	5	20	-10	22
15	7	-4	-4	-6	-1	16	-3	3	-8	3	16	-6	-4	7	19	4	4	22	-10	21
15	6	-5	-4	-5	-2	15	-4	4	-7	3	18	-6	-4	8	19	3	5	21	-11	23
15	7	-4	-4	-6	-1	17	-4	4	-8	3	17	-7	-4	7	20	4	4	22	-10	21
16	7	-5	-4	-6	-2	16	-4	4	-8	3	18	-6	-4	8	19	3	5	22	-11	23
16	7	-5	-4	-6	-2	17	-4	4	-8	3	19	-7	-5	9	20	4	4	24	-12	24
16	7	-5	-4	-6	-2	16	-4	4	-8	3	20	-7	-5	9	21	4	5	24	-12	25
17	7	-5	-5	-6	-2	17	-4	4	-8	3	19	-7	-4	8	22	4	5	22	-10	23
16	7	-5	-4	-6	-2	17	-4	4	-8	3	20	-7	-5	9	21	4	5	24	-12	25
17	7	-5	-5	-6	-2	17	-4	4	-8	3	19	-7	-4	8	22	4	5	23	-11	24
18	8	-5	-5	-7	-2	18	-4	4	-9	3	19	-7	-4	8	22	4	5	24	-11	24
18	8	-5	-5	-7	-2	19	-4	4	-9	3	20	-8	-5	9	23	5	4	26	-12	25
18	8	-5	-5	-7	-2	18	-4	4	-9	3	21	-8	-5	9	24	5	5	26	-12	26
18	8	-5	-5	-7	-2	19	-4	4	-9	3	21	-8	-5	9	24	5	5	26	-12	26

References

[1] Leiden declaration on artificial intelligence and mathematics, June 2026. Endorsed by the International Mathematical Union. `https://leidendeclaration.ai/`.

[2] David Avis and Komei Fukuda. A pivoting algorithm for convex hulls and vertex enumeration of arrangements and polyhedra. *Discrete & Computational Geometry*, 8:295–313, 1992.

[3] Igor A. Baburin and Peter Engel. On the enumeration of the combinatorial types of primitive parallelohedra in E^d, $2 \leq d \leq 6$. *Acta Crystallographica Section A: Foundations of Crystallography*, 69(5):510–516, 2013.

[4] David Bremner, Komei Fukuda, and Ambros Marzetta. Primal-dual methods for vertex and facet enumeration. *Discrete & Computational Geometry*, 20:333–357, 1998.

[5] Winfried Bruns and Bogdan Ichim. Normaliz: algorithms for affine monoids and rational cones. *Journal of Algebra*, 324(5):1098–1113, 2010.

[6] Kenneth L. Clarkson. More output-sensitive geometric algorithms. In *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, pages 695–702. IEEE Computer Society, 1994.

[7] Michel Deza and Monique Laurent. *Geometry of Cuts and Metrics*, volume 15 of *Algorithms and Combinatorics*. Springer, Berlin, 1997.

[8] Mathieu Dutour and Frank Vallentin. Some six-dimensional rigid forms. In H. Syta, A. Yurachivsky, and P. Engel, editors, *Voronoi's Impact on Modern Science, Book 3*, volume 55 of *Proceedings of the Institute of Mathematics of the National Academy of Sciences of Ukraine*, pages 102–108. Kiev, 2005. Preprint at `https://arxiv.org/abs/math/0401191`.

[9] Mathieu Dutour Sikirić. polyhedral, a GAP package. https://github.com/MathieuDutSik/GAPpackages.
[10] Mathieu Dutour Sikirić. Polytopes, lattices and quadratic forms programs. GitHub repository. Version of 19 November 2026, https://github.com/MathieuDutSik/polyhedral_common.
[11] Mathieu Dutour Sikirić. The Birkhoff polytope of the groups F_4 and H_4. In Tomislav Došlić, Snježana Majstorović, and Luka Podrug, editors, *Proceedings of the 4th Croatian Combinatorial Days*, pages 21–26, 2023. Preprint at https://arxiv.org/abs/2212.08452.
[12] Mathieu Dutour Sikirić. List of Gram matrices in the interior of iso-Delaunay of rank 6 whose extreme rays are of full rank. Zenodo, 2026. https://doi.org/10.5281/zenodo.24377749.
[13] Mathieu Dutour Sikirić, Herbert Gangl, Paul E. Gunnells, Jonathan Hanke, Achill Schürmann, and Dan Yasaki. On the cohomology of linear groups over imaginary quadratic fields. *Journal of Pure and Applied Algebra*, 220(7):2564–2589, 2016.
[14] Mathieu Dutour Sikirić, Alexey Garber, Achill Schürmann, and Clara Waldmann. The complete classification of five-dimensional Dirichlet–Voronoi polyhedra of translational lattices. *Acta Crystallographica Section A: Foundations and Advances*, 72(6):673–683, 2016.
[15] Mathieu Dutour Sikirić and Viatcheslav Grishukhin. How to compute the rank of a Delaunay polytope. *European Journal of Combinatorics*, 28(3):762–773, 2007. https://arxiv.org/abs/math/0512193.
[16] Mathieu Dutour Sikirić, Achill Schürmann, and Frank Vallentin. A generalization of Voronoi's reduction theory and its application. *Duke Mathematical Journal*, 142(1):127–164, 2008.
[17] Michael Joswig. Beneath-and-beyond revisited. In Michael Joswig and Nobuki Takayama, editors, *Algebra, Geometry, and Software Systems*, pages 1–21. Springer, 2003.
[18] Sebastian Miebach. Sphere coverings of Eisenstein, Gaussian and Hurwitzian lattices. Master's thesis, University of Cologne, 2025. Master's thesis, supervised by Frank Vallentin.
[19] Theodore S. Motzkin, Howard Raiffa, Gerald L. Thompson, and Robert M. Thrall. The double description method. In *Contributions to the Theory of Games II*, volume 28 of *Annals of Mathematics Studies*, pages 51–73. Princeton University Press, 1953.
[20] Wilhelm Plesken and Bernd Souvignier. Computing isometries of lattices. *Journal of Symbolic Computation*, 24(3–4):327–334, 1997.
[21] Achill Schürmann. Enumerating perfect forms. In *Quadratic Forms—Algebra, Arithmetic, and Geometry*, volume 493 of *Contemporary Mathematics*, pages 359–377. American Mathematical Society, Providence, RI, 2009. https://arxiv.org/abs/0901.1587.
[22] Achill Schürmann and Frank Vallentin. scc (secondary cone cruiser). Computer program.
[23] Frank Vallentin. *Sphere Coverings, Lattices, and Tilings (in Low Dimensions)*. PhD thesis, Technische Universität München, 2003. https://mediatum.ub.tum.de/doc/602017/file.pdf.
[24] Georges Voronoï. Nouvelles applications des paramètres continus à la théorie des formes quadratiques. deuxième mémoire: Recherches sur les parallélоèdres primitifs. *Journal für die reine und angewandte Mathematik*, 134:198–287, 1908.

Mathieu Dutour Sikirić, MSM programming, Karlovacka cesta 28B, 10452 Klinca selo

Email address: mathieu.dutour@gmail.com