

COMBINATORICS OF SLICES OF CUBES

MARIE-CHARLOTTE BRANDENBURG

CHIARA MERONI

Abstract

We present a complete computational classification of the combinatorial types of hyperplane sections, or slices, of the regular cube up to dimension six. For each dimension, we determine the exact number of distinct combinatorial types. When restricted to slices through the origin, our computations extend to dimension seven. The classification combines combinatorial, algebraic, and numerical techniques, with all results certified. Beyond enumeration, we analyze the distribution of types by number of vertices, establish new theoretical results about the combinatorics of slices of cubes, and propose conjectures motivated by our computational findings.

1 Introduction

The cube is a fundamental yet intricate mathematical object, discussed extensively in numerous surveys and books [Sak93; Zon05; Zon06; NT23]. In this article, we focus on *slices*, i.e., $(d - 1)$ -dimensional polytopes obtained as the intersection of the d -dimensional cube $C_d = [-1, 1]^d$ with a hyperplane. Because cubes appear throughout mathematics, their slices are likewise of broad interest, ranging across combinatorics, discrete geometry, optimization, machine learning, metric and convex geometry, number theory, and analysis. This article presents a complete computational classification of the combinatorics of slices of the cube up to dimension 6, together with a collection of theoretical results and conjectures. For this classification we combine several computational methods from combinatorics and numerical algebraic geometry, grounded in algebraic statistics and differential topology.

Theorem 1.1. *The number of combinatorial types of $(d - 1)$ -dimensional hyperplane sections of the cube C_d in dimensions $d = 3, 4, 5, 6$ is 4, 30, 344, 7346 respectively.*

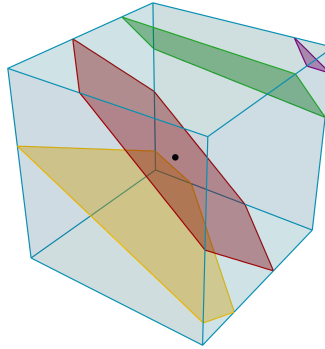


Figure 1: The cube C_3 and all its combinatorial types of slices: a purple triangle, a green quadrilateral, a yellow pentagon, a red hexagon.

When restricting to slices through the origin, our computation extends to dimension 7. Indeed, as shown below, the complexity of computing affine slices of C_d matches that of computing central slices of C_{d+1} . Our results for central slices are summarized in the following theorem.

Theorem 1.2. *The number of combinatorial types of $(d-1)$ -dimensional hyperplane sections through the origin of the cube C_d in dimensions $d = 3, 4, 5, 6, 7$ is 2, 6, 23, 133, 1657 respectively.*

Combinatorial types. The number of combinatorial types of $(d-1)$ -slices of the d -dimensional cube C_d is well known to be 4 for $d = 3$, see [Figure 1](#). For $d = 4$, Evers appears to have shown that this number is at least 30 [[Eve10](#); [FR12](#)], although the result was never published. In this article, we present exact counts for $d = 4, 5, 6$, and confirm that Evers’ upper bound is indeed tight. Independently, Nakamura and Sasaki provided a count for $d = 4$ (see [[NS80](#)]; in Japanese, not available online and not peer-reviewed), which was extended to $d = 5$ in the work of Fukuda et al. [[FMGN97](#)]. However, their numbers differ from those presented here – we reproduce their computations and explain the discrepancy, which stems from different definitions of combinatorial types.

Number of vertices. The maximum number of vertices of affine slices of C_d was determined by O’Neil, and more recently De Loera et al. identified the sequences of vertex numbers occurring among all affine slices for $d = 6$ [[O’N71](#); [DLLCT25](#)]. In this article, we extend these results by providing the number of combinatorial types corresponding to each possible vertex count, and we prove that O’Neil’s bound can be attained not only by affine slices but even by slices through the origin.

Computational methods. In order to perform these computations we combine combinatorial, algebraic and numerical techniques, supported by several software packages. [[BDLM25](#)] provides algorithms to compute combinatorics and volumes of slices for general polytopes, with implementations in SageMath [[Sag24](#)], but these are infeasible for the computationally demanding 6-dimensional cube. To overcome these limitations, we draw on the theory of the maximum-likelihood degrees of very affine varieties [[Huh13](#)], together with recent generalizations to hypersurface arrangements [[RW24](#)]. Key Julia [[BEKS17](#)] packages include the recent `HypersurfaceArrangements.jl` [[BSW24](#)], which rests on Morse theory and on `HomotopyContinuation.jl` [[BT18](#)], as well as `Oscar.jl` [[Osc](#); [DEF+25](#)] for combinatorial tasks such as intersecting and comparing polytopes. Although our approach relies on numerical methods, all results are numerically certified, ensuring that the floating-point computations correspond to exact solutions.

Outline. Given the extensive literature on slices of cubes, we conclude the introduction with a brief overview of known results. In [Section 2](#) we present computational and theoretical results on the combinatorics of slices, both affine and central. The main statements are [Theorems 2.2](#) and [2.6](#). [Section 3](#) describes the computational framework and provides additional results and conjectures (see, e.g., [Conjecture 4](#)) suggested by the data, and also contains the proofs of [Theorems 1.1](#) and [1.2](#). In [Section 4](#) we compare our computations with those of Fukuda et al. [[FMGN97](#)], leading to further open questions. Finally, we provide two appendices: [Appendix A](#) displays plots of the distributions of the combinatorial types of slices, and [Appendix B](#) describes the online repository <https://doi.org/10.5281/zenodo.17304584> where both our code and collected data are available.

Related work

Threshold functions. A key step in computing the combinatorics of cube slices is the computation of the *threshold arrangement*. A *linear threshold function* (or *linear threshold device*) is a linear function whose zero set is a hyperplane that partitions the vertices of the cube, and a self-dual linear threshold function corresponds to a hyperplane through the origin. The parameter space of linear threshold functions is subdivided into regions induced by the *threshold arrangement*, a hyperplane arrangement on which the vertex partition remains constant within each cell. This arrangement is closely tied to the task of classifying slices of the cube and its computation is therefore a crucial step of the algorithms developed in this article. Threshold functions play a fundamental role in the early theory of neural networks [Cov64; Abe77; WAP00; Ojh00] and remain relevant for modern neural network architectures [CDCBA22; EGLP23; KCB23]. Interestingly, they are also connected to topics in convex geometry and algebraic combinatorics, such as zonotopes [MM15] and triangulations of the root polytope of type A [GMP21]. Considerable computational effort has been devoted to determining the exact number of linear threshold functions (i.e., the number of cells of the threshold arrangement) for $d \leq 10$ [Mur62; Win65; MTB70; BEK23], and it is known that the logarithm of this number grows asymptotically like d^2 [Zue92].

Volumes. Extensive tools have been developed to study the volumes of slices of cubes. A classical problem concerns *extremal slices*, i.e., those with maximal or minimal volume among all slices. Hadwiger showed in the 1970s that the minimum volume of a slice through the center of symmetry is attained by a hyperplane parallel to a facet ([Had72]; in German), a result later reproved by Hensley [Hen79], who additionally conjectured the maximum volume of a slice. A few years later, Ball proved in his seminal work that the maximum-volume slice is orthogonal to a main diagonal of a 2-face [Bal86]. These results continue to influence modern research, inspiring variations such as the computation of extremal slices for normalized volumes [Ali20; BF22], the description of the space of all extremal slices [Amb22], and the study of extremal lower-dimensional slices of cubes [Vaa79; Bal87; IT21]. Beyond extremal cases, one may also consider the volume as a function over all hyperplane sections of a cube. This has lead to formulas for the volume of general $(d - 1)$ -dimensional slices [MM08; FR12] or for specific subclasses [CL91; BFGM20], and it has been shown in [BBMS22; BDLM25] that the integral of any polynomial (such as the volume) over the slices is piecewise polynomial on the cells of a hyperplane arrangement. An active line of current research studies extremal slices of cubes cut by hyperplanes at distance t from the origin [MSZZ13; Pou23a; Pou23b; AG24; Pou25], motivated by a conjecture of Milman – reported by König and Koldobsky [KK11] – on the location of such slices for specific ranges of t . Further variations include slices with maximal perimeter [Kön21], slices maximizing different measures [Zva08; KK19], and the volumes of slabs of cubes [MM08; KK11].

Related volume questions on polytopes. One way to study the volumes of all central slices of a cube, or more generally of any polytope, is through its *intersection body*, an object closely connected to the classical *Busemann–Petty problem* [BP56]. This problem asks whether, whenever the $(d - 1)$ -dimensional volumes of the slices of one centrally symmetric convex body are bounded above by those of another, the same inequality holds true for their d -dimensional volumes. A simple high-dimensional counterexample compares slices of cubes with slices of balls. Indeed, the Busemann–Petty problem holds only in dimensions up to 4 [GKS99], and the quest to establish this fact led to the development of the theory of inter-

section bodies [Lut88; Gar94; Kol98]. Since then, the study of intersection bodies of cubes has attracted independent interest; see, e.g., [Ali08; Ali20; BBMS22]. The negative resolution of the Busemann-Petty problem also inspired Bourgain’s slicing conjecture – now a theorem [KL25] – which asserts the existence of a universal constant in the inequality for d -dimensional volumes. Analogous questions can be formulated for lattice point counts [FH24].

Variations on cubes. The cube can also be seen as the unit ball of the ℓ_∞ -norm, and most of the questions discussed above have natural analogues for unit balls of ℓ_p -norms, more in general [MP88; Ole03; Kol05; Kön25].

On another note, instead of changing the norm, one can change the notion of volume by studying the *discrete volume*, that is, the number of lattice points in a slice of a polytope. This quantity is not well understood for general polytopes, but for slices of cubes both a closed formula for the number of lattice points [Abe18] and a combinatorial description of their Ehrhart polynomials [FM24] are known.

Another point of view comes from probability and concerns the number of faces of a slice. For central slices, there is a closed formula for the expected number of vertices of a random k -dimensional slice of C_d , together with a lower bound on the expected number of faces of any dimension [Lon00]. For affine slices, the expected number of vertices of a random k -dimensional slice – with a suitable distribution – is 2^k , independent of the dimension of the cube [Swa16].

Acknowledgments

We would like to thank Jesús De Loera, Simon Telen, and Kexin Wang for many fruitful conversations. We are especially grateful to Saiei-Jaeyeong Matsubara-Heo for locating the paper [NS80] and generously translating it for us. We also thank Benjamin Schröter for pointing us to [PVS22, Example 23]. CM is supported by Dr. Max Rössler, the Walter Haefner Foundation, and the ETH Zürich Foundation. MB is supported by the SPP 2458 “Combinatorial Synergies”, funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation).

2 Description of the slices

In this section, we present computational and theoretical results on slices of the centrally symmetric cube $C_d = [-1, 1]^d$. Here, a *slice* is any $(d-1)$ -dimensional polytope which can be obtained as the intersection of C_d with a hyperplane. Lower-dimensional polytopes obtained in this way are disregarded. A hyperplane is said to be *generic* with respect to C_d if it does not intersect any vertex of C_d . A slice of C_d is a *generic slice* if it arises as the intersection of C_d with a generic hyperplane. A slice of C_d is a *central slice* if the defining hyperplane contains the origin. When we wish to emphasize that the hyperplane need not contain the origin, we refer to the slice as *affine*.

The *combinatorial type* of a polytope is the isomorphism class of its face lattice. Although any full-dimensional polytope admits infinitely many slices from a metric point of view, only finitely many distinct combinatorial types occur. Counting slices according to their combinatorial type is therefore a natural and finite classification strategy. Table 1 displays the numbers of combinatorial types of slices of the cube C_d for dimensions $d \leq 7$. All results for

	$d = 2$	3	4	5	6	7
affine slices	1	4	30	344	7346	-
generic affine slices	1	4	12	58	554	-
central slices	1	2	6	23	133	1657
generic central slices	1	2	3	7	21	135

Table 1: Number of combinatorial types of $(d - 1)$ -dimensional slices of the cube $C_d = [-1, 1]^d$. For the C_7 , the affine slices computation is out of reach.

$d = 4, 5, 6, 7$ are new. We emphasize that our computations yield a complete list of all combinatorial types, whereas Table 1 merely records their counts. A complete list of representatives of each combinatorial type of slice is available at

<https://doi.org/10.5281/zenodo.17304584>

2.1 Generic central slices

When restricting to central slices, and in particular generic central slices, the rigidity of the cube can be exploited to characterize combinatorial types. Indeed, two generic central slices of the cube are combinatorially equivalent if and only if the corresponding hyperplanes induce the same partition of the sets of vertices of the cube, up to its symmetries. We now formalize this observation. A central hyperplane H is determined by its normal vector u (unique up to rescaling), and we write $H = u^\perp$. Every vector $u \in \mathbb{R}^d$ induces a partition of the vertices of C_d , grouping them into those with nonnegative scalar product with u and those with negative scalar product.

A *signed permutation* π of the set $[d]$ is a permutation of the set $\{-d, -d+1, \dots, -1, 1, \dots, d\}$ satisfying $\pi(-i) = -\pi(i)$. With a slight abuse of notation, we let π act on vectors in $\mathbb{R}^d$ by

$$\pi(v) = \pi(v_1, \dots, v_d) = (v_{\pi(1)}, \dots, v_{\pi(d)}),$$

where we define $v_{-i} = -v_i$. We denote by B_d the group of signed permutations of $[d]$.

Lemma 2.1. *For any $u \in \mathbb{R}^d$, the hyperplane $u^\perp$ either intersects all the facets of C_d (not necessarily in their relative interiors) or intersects all but two parallel facets. In the latter case, $C_d \cap u^\perp$ is combinatorially equivalent to C_{d-1} .*

Proof. Assume that $u^\perp$ does not intersect a facet of C_d , which without loss of generality we assume to be contained in the affine hyperplane $H_{+d} = \{x \in \mathbb{R}^d \mid x_d = 1\}$. By symmetry, $u^\perp$ then does not intersect the parallel facet contained in $H_{-d} = \{x \in \mathbb{R}^d \mid x_d = -1\}$. However, every edge of the cube that is not contained in one of these two facets has exactly one vertex in H_{+d} and one in H_{-d} , so each such edge meets $u^\perp$. This proves the claim. $\square$

This simple observation, relying on the rigid structure of the regular cube, allows us to reconstruct C_d (up to symmetry) from the combinatorics of a given slice.

Theorem 2.2. *Let $u_1^\perp, u_2^\perp \subset \mathbb{R}^d$ be hyperplanes generic with respect to C_d . Then, $C_d \cap u_1^\perp$ is combinatorially equivalent to $C_d \cap u_2^\perp$ if and only if there exists a signed permutation $\pi \in B_d$ such that $u_1^\perp$ and $\pi(u_2)^\perp$ intersect the same edges of C_d .*

Proof. Assume first that there exists a signed permutation $\pi \in B_d$ such that the partition of the vertices of C_d induced by u_1 coincides with the partition induced by $\pi(u_2)$. Then, by [BBMS22, Lemma 2.4], the slices $C_d \cap u_1^\perp$ and $C_d \cap \pi(u_2)^\perp$ are combinatorially equivalent. Moreover, the symmetries of the cube imply $C_d = \pi(C_d)$, and therefore $C_d \cap \pi(u_2)^\perp$ is affinely isomorphic, and hence combinatorially equivalent, to $C_d \cap u_2^\perp$. This proves one direction of the statement (note that we did not use the fact that the slices are central).

Conversely, assume that $C_d \cap u_1^\perp$ is combinatorially equivalent to $C_d \cap u_2^\perp$. By the genericity of these hyperplanes, Theorem 2.1 implies that $C_d \cap u_1^\perp$ has either $2(d-1)$ facets or $2d$ facets. If $C_d \cap u_1^\perp$ has $2(d-1)$ facets, then for each $i = 1, 2$, the hyperplane $u_i^\perp$ partitions the vertices of C_d into those lying in the facets

$$\{x \in \mathbb{R}^d \mid x_{j_i} = 1\} \quad \text{and} \quad \{x \in \mathbb{R}^d \mid x_{j_i} = -1\}.$$

Let $\pi \in B_d$ be any signed permutation satisfying $\pi(j_2) = j_1$. It follows that $\pi(u_2)^\perp$ partitions the vertices of C_d inside $\{x \in \mathbb{R}^d \mid x_{j_1} = 1\}$ and $\{x \in \mathbb{R}^d \mid x_{j_1} = -1\}$ exactly as $u_1^\perp$ does, implying that $u_1^\perp$ and $\pi(u_2)^\perp$ intersect C_d in the same set of edges in their relative interiors.

It remains to consider the case where $C_d \cap u_1^\perp$ has $2d$ facets. We label the facets of C_d as

$$F_{\star j} = C_d \cap \{x \in \mathbb{R}^d \mid x_j = \star 1\}, \quad \star \in \{+, -\}.$$

Let M_1 and M_2 be the vertex–facet incidence matrices of the two slices, each with $2d$ columns indexed by $[d] \cup -[d]$. Since the slices are combinatorially equivalent, there exist permutations π of the columns and $\sigma : [n] \rightarrow [n]$ of the rows such that $(M_1)_{\sigma(a), \pi(b)} = (M_2)_{a, b}$. Because both slices are centrally symmetric, π maps every pair of opposite integers $(k, -k)$ to another pair of opposite integers $(\pi(k), \pi(-k)) = (\ell, -\ell)$, so $\pi \in B_d$ indeed maps the columns of M_2 to those of M_1 .

Without loss of generality we may assume $\sigma = \text{id}$. Suppose that $\pi(u_2)^\perp$ intersects an edge e of C_d . Then $e \cap \pi(u_2)^\perp$ is a vertex of $C_d \cap \pi(u_2)^\perp$, corresponding to the k^{th} row of M_2 . From $(M_2)_{k, \star j} = (M_1)_{k, \pi(\star j)}$ we deduce

$$e = \bigcap_{\substack{\star j \in [d] \cup -[d] \\ (M_2)_{k, \star j} = 1}} F_{\pi(\star j)} = \bigcap_{\substack{\star j \in [d] \cup -[d] \\ (M_1)_{k, \pi(\star j)} = 1}} F_{\pi(\star j)}.$$

The rightmost expression is precisely the edge intersected by $u_1^\perp$ to produce the k^{th} vertex of $C_d \cap u_1^\perp$. This holds for every edge of the slices, showing that $u_1^\perp$ and $\pi(u_2)^\perp$ intersect C_d in exactly the same set of edges. $\square$

Example 2.3. Let $u_1 = \frac{1}{\sqrt{3}}(1, 1, 1)$ and $u_2 = \frac{4}{\sqrt{29}}(1, -\frac{1}{2}, -\frac{3}{4})$. The corresponding slices are both hexagons, as shown in Figure 2, with vertex–facet incidence matrices equal (up to permutation) to

$$M_1 = M_2 = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

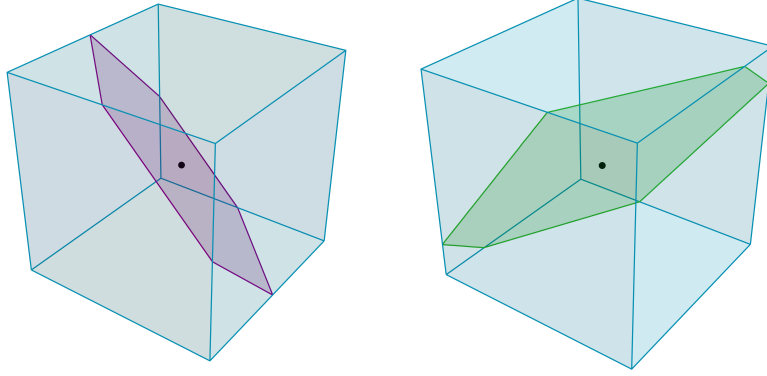


Figure 2: The slices $C_3 \cap u_1^\perp$ (left) and $C_3 \cap u_2^\perp$ (right) from [Theorem 2.3](#).

The columns of M_1 are indexed by $F_{+1}, F_{-2}, F_{+3}, F_{-1}, F_{+2}, F_{-3}$, and the columns of M_2 are indexed by $F_{+1}, F_{+2}, F_{-3}, F_{-1}, F_{-2}, F_{+3}$. This yields the signed permutation $\pi \in B_3$ given by

$$1 \mapsto 1, \quad 2 \mapsto -2, \quad 3 \mapsto -3.$$

The vectors u_1 and $\pi(u_2) = \frac{4}{\sqrt{29}}(1, \frac{1}{2}, \frac{3}{4})$ induce the same partition of the vertices of C_3 into the two sets $\pm\{(1, 1, 1), (1, 1, -1), (1, -1, 1), (-1, 1, 1)\}$. Hence, the hyperplanes $u_1^\perp$ and $\pi(u_2)^\perp$ intersect exactly the same set of edges of C_3 . $\diamond$

Despite one direction of [Theorem 2.2](#) not requiring the slices to be central (see first paragraph of its proof), the converse direction – reconstructing the cube from a slice – cannot be extended to noncentral slices, as the following example shows.

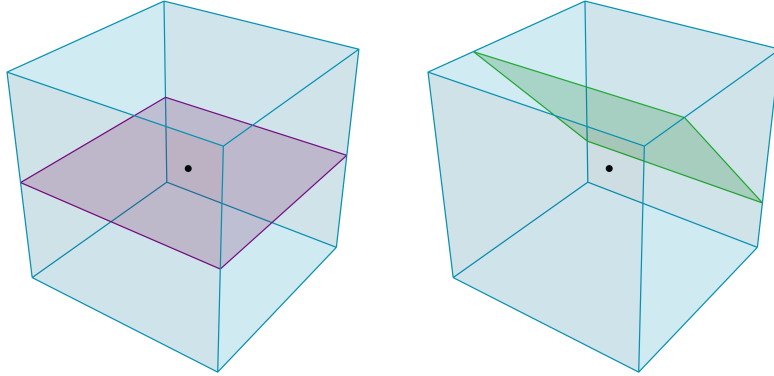


Figure 3: The slices $C_3 \cap H_1$ (left) and $C_3 \cap H_2$ (right) from [Theorem 2.4](#).

Example 2.4. Let $H_1 = \{z = 0\}$ and $H_2 = \{2y + 2z = 1\}$. The corresponding slices are both quadrilaterals, as displayed in [Figure 3](#), with vertex-facet incidence matrices equal (up to permutation) to

$$M_1 = M_2 = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{pmatrix}.$$

The columns of M_1 are indexed by $F_{+1}, F_{+2}, F_{-1}, F_{-2}$, whereas those of M_2 are indexed by $F_{+1}, F_{+2}, F_{-1}, F_{+3}$. This defines a (partial) permutation of $\{\pm 1, \pm 2, \pm 3\}$ sending

$$1 \mapsto 1, \quad 2 \mapsto 2, \quad -1 \mapsto -1, \quad -2 \mapsto 3.$$

Such a mapping cannot be extended to a signed permutation, and indeed there is no signed permutation under which the two hyperplanes intersect the same set of edges of C_3 . $\diamond$

This relation between the combinatorics of generic central slices of the cube and hyperplanes that partition its vertices leads naturally to the notion of threshold functions. A *threshold function* on d (or fewer) variables is a function

$$f_{(w_0, w)} : \{-1, 1\}^d \rightarrow \{0, 1\}, \quad f_{(w_0, w)}(x) = \begin{cases} 1 & \text{if } \langle w, x \rangle \geq w_0 \\ 0 & \text{otherwise,} \end{cases}$$

where $(w_0, w) \in \mathbb{R}^{d+1}$. After a small perturbation of (w_0, w) we may assume that $\langle w, x \rangle \neq w_0$ for all $x \in \{-1, 1\}^d$. Geometrically, a threshold function $f_{(w_0, w)}$ records, for each vertex of the cube C_d , on which side of the hyperplane $\{x \in \mathbb{R}^d \mid \langle w, x \rangle = w_0\}$ the vertex lies.

The complement of a variable x_i is $-x_i$. Threshold functions admit an $(S_2)^d$ -action by independently complementing each variable, and an additional (S_d) -action by permuting the coordinates. An *NP-equivalence class* of a threshold function is its equivalence class under these combined group actions. In other words, NP-equivalence classes correspond to orbits under the group B_d of signed permutations. A threshold-function is *self-dual* if it takes opposite values on complementary inputs, i.e., $f(-x) = 1 - f(x)$ for all $x \in \{-1, 1\}^d$. For this to hold, we must have

$$\langle w, -x \rangle \geq w_0 \iff \langle w, x \rangle < w_0,$$

which is equivalent to

$$w_0 + \langle w, x \rangle \leq 0 \text{ and } w_0 - \langle w, x \rangle > 0 \quad \text{for all } x \in \{-1, 1\}^d.$$

This can occur only if $w_0 = 0$. Thus every self-dual threshold function corresponds geometrically to a hyperplane through the origin. Combining this with [Theorem 2.2](#) yields the following corollary.

Corollary 2.5. *Combinatorial types of generic central slices of the cube C_d are in bijection with NP-equivalence classes of self-dual threshold functions on d variables.*

2.2 Vertices of slices

In [\[DLLCT25\]](#), the authors give a complete characterization of the possible numbers of vertices of hyperplane sections of C_d (regardless of the dimension of the intersection) for $d \leq 7$. We confirm their numbers and additionally determine which of these are realized as vertices of full-dimensional hyperplane sections, namely as vertices of what we call *slices*. A natural question is the distribution of these vertex numbers: how many combinatorial types of slices have a fixed number of vertices? This is shown in [Figures 4 and 5](#), which display these distributions for various types of slices of C_4 , C_5 , and C_6 . Here, the light green bars represent

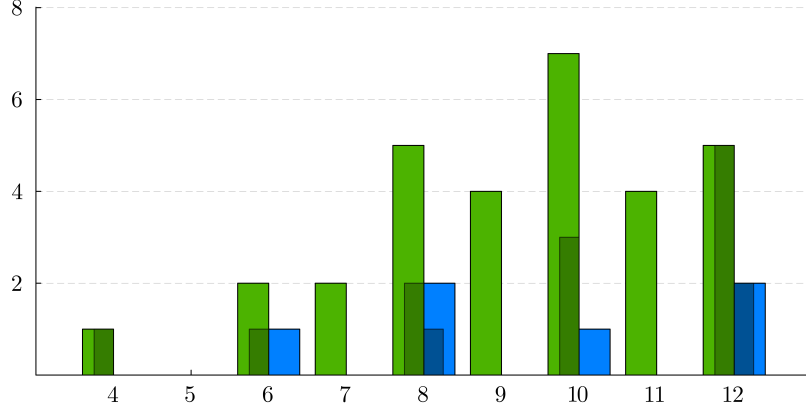


Figure 4: Number of combinatorial types of slices of C_4 by number of vertices.

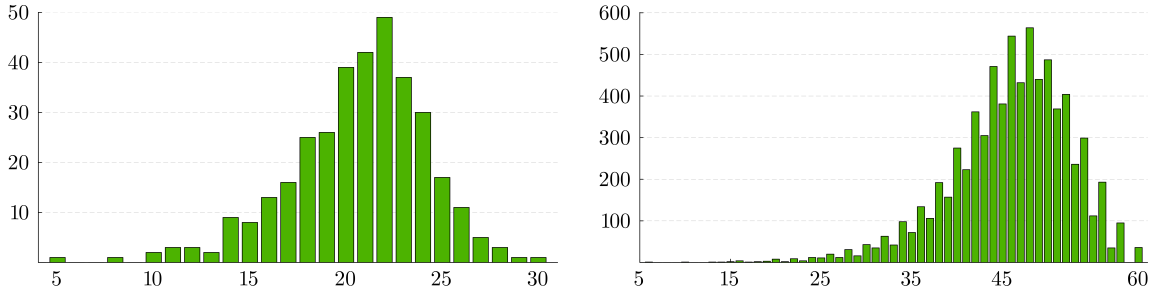


Figure 5: Number of combinatorial types of affine slices of C_5 and C_6 by number of vertices.

all affine slices, the dark green bars represent generic affine slices, and the light and dark blue bars correspond to central and generic central slices, respectively.

Moreover, [DLLCT25, Theorem 1.4] shows that there are gaps in the possible numbers of vertices of slices of cubes. However, the authors also allow polytopes of dimension strictly lower than $d - 1$, obtained as intersections of C_d with a hyperplane. This relaxation yields a larger set of vertex numbers. Our computations indicate that some of these are attained *only* by lower-dimensional sections. This is evident for hyperplane sections with fewer than $d + 1$ vertices (by a simple dimension count), but these are not the only cases. These observations motivate the following conjecture, which we have verified computationally for $d \leq 6$.

Conjecture 1. *There are no $(d - 1)$ -dimensional slices of the cube C_d with 2^i vertices, for any i satisfying $2^i \leq 2d - 3$.*

Let us now focus on the maximum possible number of vertices of a slice of a cube. O’Neil showed that this number is at most $\lceil \frac{d}{2} \rceil \binom{d}{\lceil \frac{d}{2} \rceil}$ [O’N71] and constructed an explicit generic affine slice attaining this bound. Our computations, summarized in Figures 4 and 5, and detailed further in Appendix A, indicate that this bound can be attained not only by generic affine slices but also by generic *central* slices – a much smaller class. Indeed, this phenomenon occurs in every dimension.

Theorem 2.6. *The upper bound $\lceil \frac{d}{2} \rceil \binom{d}{\lceil \frac{d}{2} \rceil}$ on the number of vertices of a $(d-1)$ -dimensional slice of C_d is attained by a generic central slice for every d .*

Proof. Every edge of the cube is parallel to a coordinate directions e_i for some $i \in [d]$. A point on such an edge can be written as $v_i(\lambda) \in \mathbb{R}^d$, where $(v_i(\lambda))_j \in \{-1, 1\}$ for $j \in [d] \setminus \{i\}$, and $(v_i(\lambda))_i = \lambda \in [-1, 1]$. Given a vector $u \in \mathbb{R}^d$, associate to each vertex of $C_d \cap u^\perp$ of the form $v_i(\lambda)$ the sets of “plus” and “minus” indices

$$\mathbf{p}(u) = \{j \in [d] \setminus \{i\} \mid (v_i(\lambda))_j = 1, u_j \neq 0\}, \quad \mathbf{m}(u) = \{j \in [d] \setminus \{i\} \mid (v_i(\lambda))_j = -1, u_j \neq 0\}.$$

First suppose that d is odd and take $u = (1, \dots, 1)$. The equation $\langle u, v_i(\lambda) \rangle = 0$ reads $|\mathbf{p}(u)| - |\mathbf{m}(u)| + \lambda = 0$, where $|\cdot|$ denotes the cardinality of a set, so $\lambda \in \{-1, 0, 1\}$. Because $\mathbf{p}(u) \sqcup \mathbf{m}(u) \sqcup \{i\} = [d]$, we get

$$d = |\mathbf{p}(u)| + |\mathbf{m}(u)| + 1 = 2|\mathbf{m}(u)| + 1 - \lambda \iff 2|\mathbf{m}(u)| = d - 1 + \lambda.$$

Since $d - 1$ is even, it follows that $\lambda = 0$ and consequently $|\mathbf{p}(u)| = |\mathbf{m}(u)| = \frac{d-1}{2}$. In other words, $u^\perp$ does not contain any vertex of C_d , and the corresponding slice is generic. For each $i \in [d]$, there are $\binom{d-1}{(d-1)/2}$ choices for $\mathbf{p}(u)$, which uniquely determines $\mathbf{m}(u)$. The total number of vertices $v_i(\lambda)$ of the slice $C_d \cap u^\perp$ is thus

$$d \binom{d-1}{\frac{d-1}{2}} = \frac{d+1}{2} \binom{d}{\frac{d+1}{2}}.$$

Now let d be even and take $u = (1, \dots, 1, 0)$. Then, $0 = \langle u, v_i(\lambda) \rangle = |\mathbf{p}(u)| - |\mathbf{m}(u)| + |\{i\} \setminus \{d\}| \lambda$, and $[d] = \mathbf{p}(u) \sqcup \mathbf{m}(u) \sqcup \{i, d\}$. If $i = d$, then $v_d(\lambda)$ is a vertex of the slice $C_d \cap u^\perp$ if and only if $|\mathbf{p}(u)| = |\mathbf{m}(u)|$. However, $d = |\mathbf{p}(u)| + |\mathbf{m}(u)| + 1 = 2|\mathbf{p}(u)| + 1$ contradicts the assumption that d is even. Thus, $u^\perp$ does not intersect edges of C_d in direction e_d . If $i \neq d$, then $v_d(\lambda)$ is a vertex of the slice if and only if $|\mathbf{p}(u)| - |\mathbf{m}(u)| + \lambda = 0$, so $\lambda \in \{-1, 0, 1\}$. Because $\mathbf{p}(u) \sqcup \mathbf{m}(u) \sqcup \{i, d\} = [d]$, we get

$$d = |\mathbf{p}(u)| + |\mathbf{m}(u)| + 2 = 2|\mathbf{m}(u)| + 2 - \lambda \iff 2|\mathbf{m}(u)| = d - 2 + \lambda.$$

Since $d - 2$ is even, we have $\lambda = 0$ and $|\mathbf{p}(u)| = |\mathbf{m}(u)| = \frac{d-2}{2}$. As in the previous case, this implies that $u^\perp$ does not contain any vertex of C_d , so the slice is generic. For each $i \in [d-1]$, there are $\binom{d-2}{(d-2)/2}$ choices for $\mathbf{p}(u)$, which uniquely determines $\mathbf{m}(u)$. The d^{th} coordinate can be chosen as $+1$ or -1 . The total number of vertices $v_i(\lambda)$ of the slice $C_d \cap u^\perp$ is thus

$$2(d-1) \binom{d-2}{\frac{d-2}{2}} = \frac{d}{2} \binom{d}{\frac{d}{2}}. \quad \square$$

Theorem 2.6 shows that for any $(d-1)$ -dimensional affine subspace, the number of vertices of its intersection with the cube is maximized by restricting to intersections with generic linear subspaces. Preliminary computations suggest that an analogous phenomenon holds for intersections with k -dimensional subspaces.

Conjecture 2. *The upper bound on the number of vertices of a k -dimensional slice of C_d can be attained by a generic central k -dimensional slice for every d and every $k \leq d-1$.*

Observe that any centrally symmetric k -dimensional polytope can be realized as the intersection of some d -dimensional cube with a k -dimensional linear subspace, for a suitable $d \geq k$. Consequently, the previous conjecture would follow from the stronger statement below.

Conjecture 3. *For every centrally symmetric k -dimensional polytope P , the upper bound on the number of vertices of a $(k - 1)$ -dimensional slice of P is attained by a central slice.*

We conclude this section by strengthening [DLLCT25, Lemma 2.1] and showing that for any polytope, a slice with the largest possible number of vertices must be generic with respect to that polytope. This applies in particular to cubes, but we state and prove it in full generality.

Proposition 2.7. *Let $P \subset \mathbb{R}^d$ be a full-dimensional polytope with $d \geq 3$. If $P \cap H$ is a $(d - 1)$ -dimensional affine slice having the largest possible number of vertices among all such slices of P , then H does not contain any vertex of P .*

Proof. The strategy is to show that any slice $P \cap H$ containing a certain number of vertices of P can be perturbed slightly so that H contains exactly one vertex of P , without diminishing the number of vertices of the slice. If we then prove that, for a slice containing exactly one vertex of P , there exists another slice with strictly more vertices, the claim follows. We start from this latter statement for simplicity.

Assume that the affine hyperplane H contains exactly one vertex of P . Without loss of generality, this vertex is the origin and $H = u^\perp$ for some $u \in \mathbb{R}^d$. Assume that the edges of P incident to the origin are

$$[0, v_i] \quad \text{for } i = 1, \dots, k$$

for some $k \in \mathbb{N}$. Since the slice $P \cap u^\perp$ has dimension $d - 1$ and contains no other vertex of P , the hyperplane $u^\perp$ cannot be supporting P . Therefore, it induces a nontrivial partition (i.e., both sets have at least one element) of the vertices $v_1, \dots, v_k$ according to the sign of $\langle u, v_i \rangle$. Because $k \geq d \geq 3$, one part has at least two elements, and without loss of generality, let this be the set with positive inner product. In other words, there exists $I \subset [k]$ with $|I| \geq 2$ such that $\langle u, v_i \rangle > 0$ for all $i \in I$. Choose $\varepsilon > 0$ such that

$$\varepsilon < \min\{\langle u, v \rangle \mid \langle u, v \rangle > 0, v \in \text{vert}(P)\},$$

and consider the parallel hyperplane $H_\varepsilon = \{x \in \mathbb{R}^d \mid \langle u, x \rangle = \varepsilon\}$. Then, the number of vertices of the associated slice satisfies

$$|\text{vert}(P \cap H_\varepsilon)| = |\text{vert}(P \cap u^\perp)| - 1 + |I| > |\text{vert}(P \cap u^\perp)|.$$

Now assume that the hyperplane H contains more than one vertex of P . Without loss of generality, one of these vertices is the origin and $H = u^\perp$. Since P is d -dimensional, at least one of the halfspaces defined by $u^\perp$ contains other vertices of P ; assume this is the side $\{\langle u, x \rangle > 0\}$. Let $v_1, \dots, v_k$ be the remaining vertices of P lying in $u^\perp$. Choose $u' \in \mathbb{R}^d$ so that $\{\langle u', v_i \rangle \geq 0\}$ for all $i = 1, \dots, k$. Because there are vertices of P in $\{\langle u, x \rangle > 0\}$, each v_i is the endpoint of at least one edge contained in $\{\langle u, x \rangle \geq 0\}$ and not in $\{\langle u, x \rangle = 0\}$. Pick $\varepsilon > 0$ such that

$$\varepsilon < \min\{\alpha > 0 \mid \langle u - \alpha u', v \rangle > 0, \langle u, v \rangle > 0, v \in \text{vert}(P)\},$$

and consider the rotated hyperplane $(u - \varepsilon u')^\perp$. By construction, this hyperplane intersects every edge of the form $[v_i, w]$ with $w \in \text{vert}(P)$ and $\langle u, w \rangle > 0$. Indeed,

$$0 = \langle u - \varepsilon u', (1 - \lambda)v_i + \lambda w \rangle = 0 + (1 - \lambda)\varepsilon\langle u', v_i \rangle + \lambda\langle u - \varepsilon u', w \rangle$$

is satisfied by

$$\lambda = \frac{\varepsilon\langle u', v_i \rangle}{\varepsilon\langle u', v_i \rangle + \langle u - \varepsilon u', w \rangle} \in [0, 1].$$

Let $E_i \geq 1$ be the number of such edges of P contained in $\{x \in \mathbb{R}^d \mid \langle u, x \rangle \geq 0\}$ and having v_i as endpoint. Then

$$|\text{vert}(P \cap (u - \varepsilon u')^\perp)| = |\text{vert}(P \cap u^\perp)| - k + \sum_{i=1}^k E_i \geq |\text{vert}(P \cap u^\perp)|. \quad \square$$

This result agrees with our cube computations: slices with the largest number of vertices must be generic. For cubes even more is true, namely that central slices with the largest number of vertices must be generic central slices. This follows directly from [Theorem 2.7](#) together with [Theorem 2.6](#). Moreover, a minor adaptation of the proof of [Theorem 2.7](#) shows that when P is centrally symmetric, the central slice (i.e., one containing the center of symmetry) with largest possible number of vertices must be a generic central slice.

3 Exact and certified numerical algorithms

In this section we examine the algorithms used to compute the data discussed above. We describe two main routines, which we will compare, along with several preprocessing steps. These algorithms apply to general polytopes, not only to cubes, so we present them in full generality. The setting differs slightly depending on whether we consider central or affine slices. When a d -dimensional polytope P is embedded in $\mathbb{R}^d$ with its center at the origin, all central slices are obtained by intersecting P with hyperplanes through the origin. In contrast, to parametrize all affine slices of P , we embed P into $\mathbb{R}^{d+1}$ at height 1, namely in the hyperplane $\{x \in \mathbb{R}^{d+1} \mid x_{d+1} = 1\}$. In this representation, every affine slice of P arises as the intersection with a hyperplane through the origin of $\mathbb{R}^{d+1}$.

Our approach follows naturally from [\[BDLM25\]](#), but we include it here for completeness. Let P be a d -dimensional polytope and let A be the matrix whose columns are the vertices of P . We consider the hyperplane arrangement

$$\mathcal{H}_A = \mathcal{H}_P = \{v^\perp \mid v \in \text{cols}(A) = \text{vert}(P)\}. \quad (1)$$

Since P is d -dimensional, each hyperplane section $P \cap u^\perp$ has dimension at most $(d - 1)$. By [\[BMS22, Lemma 2.4\]](#), if u_1, u_2 belong to the same cell of the hyperplane arrangement $\mathcal{H}_P$ in (1), then they induce the same partition of the vertices of P , and hence the slices $P \cap u_1^\perp$ and $P \cap u_2^\perp$ have the same combinatorial type. Therefore, to enumerate all combinatorial types of slices of P , it suffices to collect one representative point per cell of $\mathcal{H}_P$ and compute the combinatorial type of the corresponding slice.

The first step – collecting one representative point per cell – encodes the main computational complexity of the algorithm, and we develop two different approaches for it. We refer to the

first as the *exact algorithm* which is outlined in [Algorithm 1](#). This method computes the full cell decomposition of the hyperplane arrangement (see [\[KP20\]](#) for the implementation in `Polymake`). Once the decomposition is obtained, since all relevant cells of $\mathcal{H}_P$ are unbounded, for each cell it suffices to take the sum of its rays to obtain a point in its interior.

Algorithm 1 exact algorithm

INPUT: Polytope P

OUTPUT: One point per cell of $\mathcal{H}_P$

```

1: construct the central hyperplane arrangement  $\mathcal{H}_P$ 
2: all_points  $\leftarrow$  empty
3: for cells of  $\mathcal{H}_P$  do
4:   compute the rays of the cell
5:   sum the rays to get a point  $p$  in the relative interior of the cell
6:   append  $p$  to all_points
7: end for
return all_points

```

The computational complexity of [Algorithm 1](#) is polynomial in the number of vertices of P for fixed dimension d , but grows exponentially with d in general. This follows from [\[GS93\]](#) and is also explained in [\[BDLM25, Section 4\]](#). For instance, the first column of [Table 2](#) shows the number of maximal cells of $\mathcal{H}_{C_d}$ for $d \leq 9$. Therefore, even when P has relatively few vertices, in higher dimension (where “high” in this scenario means at least five or six) computing the exact decomposition of $\mathcal{H}_P$ quickly becomes infeasible.

To extend the computation further, we introduce a second method, which we refer to as the *certified numerical algorithm*. The underlying theory goes back to [\[Var95; Huh13\]](#) and has recently been generalized in [\[RW24\]](#). For algorithmic aspects, see also [\[BSW24\]](#). For the convenience of the reader, we describe the details of the algorithm below, omitting proofs.

Let $\ell_1, \dots, \ell_n$ be the linear forms associated with the vertices of P , namely $\ell_i(x) = \langle v_i, x \rangle$, for $\text{vert}(P) = \{v_1, \dots, v_n\}$. Consider the function

$$\psi(x) = \sum_{i=1}^n u_i \log |\ell_i(x)| - v \log |g(x)|, \quad (2)$$

where g is a generic quadratic polynomial with $g(x) > 0$, and $u_i, v \in \mathbb{R}$ are some parameters. By [\[RW24, Theorem 1.2\]](#), if v is sufficiently large (specifically, if $2v \geq \sum_{i=1}^n u_i$), then the number of critical points of ψ equals the number of connected components of the complement of $\cup_{i=1}^n \{\ell_i(x) = 0\}$. Moreover, each critical point of ψ lies in exactly one such component. Computing these critical points amounts to solving a system of polynomial equations. This can be done efficiently with classical methods from numerical algebraic geometry, such as `HomotopyContinuation.jl` [\[BT18\]](#). Despite being numerical, this approach allows one to *certify* the computed solutions (see [Theorem 3.4](#)).

[Algorithm 2](#) corresponds to [\[RW24, Algorithm 1\]](#). Strictly speaking, in order to find the zeros of the gradient one must apply monodromy methods to solve the resulting system of rational equations. Since this is not the focus of the present paper, we omit these details and refer to [\[RW24\]](#). When $P = C_d$, the choice $g(x) = \sum_i x_i^2 + 1$ is sufficiently generic, see [Section 3.1](#).

Algorithm 2 sample points in maximal cells of $\mathcal{H}_A$

INPUT: Matrix A with columns $A_1, \dots, A_n$

OUTPUT: One point in each maximal cell of $\mathcal{H}_A$

1: $\psi \leftarrow \sum_{i=1}^n u_i \log |\langle A_i, x \rangle| - v \log |g(x)|$

2: $\text{sols} \leftarrow \text{solve}(\nabla_x \psi = 0)$

return sols

This algorithm alone does not fully solve our problem, since it produces points only in the *maximal* cells of the hyperplane arrangement $\mathcal{H}_P = \mathcal{H}_A$. To also collect points in the lower-dimensional cells, we iteratively restrict to intersections of hyperplanes in the arrangement and apply [Algorithm 2](#) in that restricted setting. Concretely, to restrict to $v_1^\perp$, where v_1 is the first column of A , we define the matrix $A|_{v_1}$ as

$$(A|_{v_1})_{i,j} = A_{i,j} - (v_1)_i \frac{A_{k,j}}{(v_1)_k} \quad \text{for } i \neq k, j = 2, \dots, |\text{vert}(P)|, \quad (3)$$

for any index k with $(v_1)_k \neq 0$. In practice, we simply pick the first such index. To restrict further, for example to $v_1^\perp \cap v_2^\perp$, we repeat the same procedure on the new matrix $A|_{v_1}$.

Remark 3.1. The restricted matrix $A|_{\mathfrak{v}}$, for a subset $\mathfrak{v} \subset \text{vert}(P)$, corresponds to restricting the hyperplane arrangement $\mathcal{H}_A$ to the face $\cap_{v \in \mathfrak{v}} v^\perp$ whose codimension equals $\dim(\text{span } \mathfrak{v})$. From the perspective of slices, points in this face of the arrangement represent normals to slices of the cube that contain all vertices in $\mathfrak{v}$.

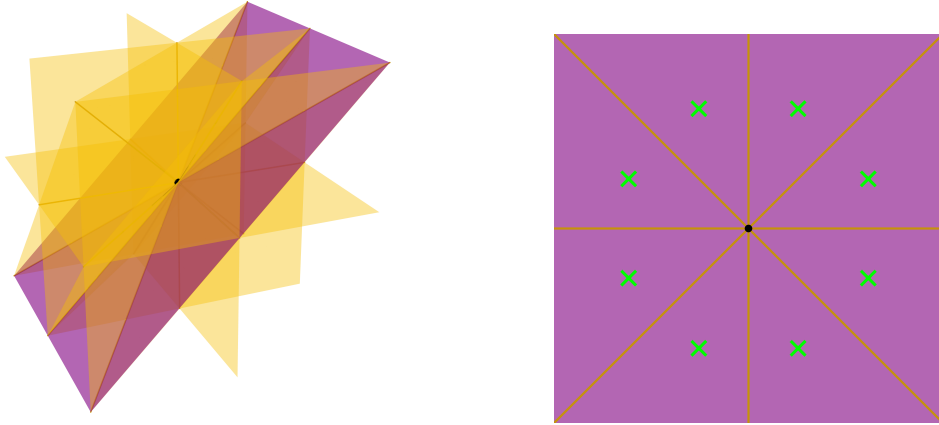


Figure 6: Hyperplane arrangements $\mathcal{H}_{A|_{v_1}}$ (left), $\mathcal{H}_{A|_{\mathfrak{v}}}$ (right) from [Theorem 3.2](#).

Example 3.2 (Restriction to subspaces). Let $P = C_3 \subset \mathbb{R}^4$ be embedded at height $x_4 = 1$, and consider the associated matrix

$$A = \begin{pmatrix} v_1 & v_2 & v_3 & v_4 & v_5 & v_6 & v_7 & v_8 \\ -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 \\ -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \end{matrix}$$

with the vertices of C_3 as columns. For $v_1 = (-1, -1, -1, 1)$ the restriction matrix is

$$A|_{v_1} = \begin{pmatrix} v_2 & v_3 & v_4 & v_5 & v_6 & v_7 & v_8 \\ -2 & 2 & 0 & 0 & -2 & 2 & 0 \\ -2 & 0 & -2 & 2 & 0 & 2 & 0 \\ 2 & 0 & 2 & 0 & 2 & 0 & 2 \end{pmatrix} \begin{matrix} 2 \\ 3 \\ 4 \end{matrix}. \quad (4)$$

The associated hyperplane arrangement $\mathcal{H}_{A|_{v_1}} \subset \mathbb{R}^3 = v_1^\perp$ is shown in [Figure 6](#), left. Suppose now we are interested in the tuple $\mathfrak{v} = \{v_1, v_4\}$. In this case, we need to iterate the restriction procedure. The hyperplane of $\mathcal{H}_{A|_{v_1}}$ corresponding to v_4 is highlighted in purple. Note that we cannot apply formula (3) with $k = 1$, since this would involve division by zero in the transformed vector v_4 , namely the third column of $A|_{v_1}$. Instead, choosing $k = 2$ yields

$$A|_{\mathfrak{v}} = \left((A|_{v_1})_{i,j} - (A|_{v_1})_{i,3} \frac{(A|_{v_1})_{2,j}}{(A|_{v_1})_{2,3}} \right)_{i,j} = \begin{pmatrix} v_2 & v_3 & v_5 & v_6 & v_7 & v_8 \\ -2 & 2 & 0 & -2 & 2 & 0 \\ 0 & 0 & 2 & 2 & 2 & 2 \end{pmatrix} \begin{matrix} 2 \\ 4 \end{matrix}. \quad (5)$$

The hyperplane arrangement $\mathcal{H}_{A|_{\mathfrak{v}}} \subset \mathbb{R}^2$, lying in the purple two-dimensional plane $v_1^\perp \cap v_4^\perp$, is displayed in [Figure 6](#), right. $\diamond$

d	0	1	2	3	4	5	6
3	104	32	6, 8	2			
4	1882	370	32, 60	6, 8, 10	2		
5	94572	11292	370, 1024 1296	32, 60, 96, 98, 128	6, 8, 10, 12, 14, 16	2	
6	15028134	1066044	11292, 47900, 47900, 73632	370, 1024, 1296, 2258, 2640, 3790, 5040	32, 60, 96, 98, 128, 144, 146, 180, 200, 220, 264, 288, 312, 336	6, 8, 10, 12, 14, 16, 18, 20, 22, 24	2
7	8378070864	347326352					
8	17561539552946	419172756930					
9	144130531453121108	1955230985997140					

Table 2: Possible numbers of maximal cells in the hyperplane arrangement $\mathcal{H}_{A|_{\mathfrak{v}}}$ in $\mathbb{R}^{d+1-\dim(\text{span } \mathfrak{v})}$ for $P = C_d \times \{1\} \subset \mathbb{R}^{d+1}$ or $P = C_{d+1} \subset \mathbb{R}^{d+1}$. The first column of the table fixes d , the other columns denote $\dim(\text{span } \mathfrak{v})$.

We collect in [Table 2](#) the numbers of maximal cells in the hyperplane arrangements $\mathcal{H}_{A|_{\mathfrak{v}}}$ for all subsets $\mathfrak{v} \subset \text{vert}(C_d)$, for $d \leq 6$, in the setting of affine slices, namely for $C_d \subset \mathbb{R}^{d+1}$. As explained in [Section 2.1](#), the maximal cells of $\mathcal{H}_A$ are in bijection with threshold functions on d (or fewer) variables, and the first column of [Table 2](#) reports the corresponding numbers, according to the OEIS sequence [A000609](#). The second column displays the *resonance*

arrangement from [BEK23, Section 6.3], which coincides with OEIS sequence A034997. This can be seen by considering the cube $[0, 1]^d$ instead of $[-1, 1]^d$, and taking $\mathbf{v} = \{(0, \dots, 0)\}$.

Once we have the restricted matrix $A|_{\mathbf{v}}$, we compute the critical points of the associated function ψ using Algorithm 2. This yields a list of points in $\mathbb{R}^m$, lying inside the linear space $\cap_{v \in \mathbf{v}} v^\perp$, where $m + \dim(\text{span } \mathbf{v})$ is the dimension of the ambient space. To express these points in the original coordinates of the ambient space, we must invert the restriction procedure.

Consider $A|_{\mathbf{v}}$ and $A|_{\mathbf{v} \cup v_\ell}$, with $\dim(\text{span}(\mathbf{v} \cup v_\ell)) = \dim(\text{span } \mathbf{v}) + 1$, and let $\tilde{v}_\ell$ be the column of $A|_{\mathbf{v}}$ corresponding to v_ℓ . Let k be the index of the row removed when going from $A|_{\mathbf{v}}$ to $A|_{\mathbf{v} \cup v_\ell}$, and let $p \in \mathbb{R}^m$ be a point in $(\cap_{v \in \mathbf{v}} v^\perp) \cap v_\ell$. We map p to the point $\hat{p} = (p_{i_1}, \dots, p_k, \dots, p_{i_m})$, $i_1 < \dots < k < \dots < i_m$, such that

$$\langle (A|_{\mathbf{v}})_{v_\ell}, \hat{p} \rangle = 0.$$

We repeat this step iteratively to recover the missing coordinates, until we get back to ambient space. All of the above steps are summarized in Algorithm 3.

Algorithm 3 certified numerical algorithm

INPUT: Polytope P

OUTPUT: One point per cell of $\mathcal{H}_P$

```

1: all_points  $\leftarrow$  empty
2:  $A \leftarrow$  matrix with columns the vertices of  $P$ 
3: sols  $\leftarrow$  Algorithm 2( $A$ )
4: append sols to all_points
5: for  $\mathbf{v}$  tuple of vertices of  $P$  do
6:   construct the matrix  $A|_{\mathbf{v}}$ 
7:   sols_lower  $\leftarrow$  Algorithm 2( $A|_{\mathbf{v}}$ ), the solutions in  $\mathbb{R}^{d+1-\dim(\text{span } \mathbf{v})}$ 
8:   sols_v  $\leftarrow$  embedding of  $\text{sols\_lower} \subset \cap_{v \in \mathbf{v}} v^\perp$  into  $\mathbb{R}^{d+1}$ 
9:   append sols_v to all_points
10: end for
    return all_points

```

We illustrate steps 7 and 8 of Algorithm 3 in a low-dimensional setting.

Example 3.3 (Steps 7 and 8 of Algorithm 3). Let $P = C_3 \subset \mathbb{R}^4$, let $\mathbf{v} = \{v_1, v_4\} = \{(-1, -1, -1, 1), (1, 1, -1, 1)\}$ as in Theorem 3.2, and consider the restricted matrix $A|_{\mathbf{v}}$ from (5). In step 7 of Algorithm 3, we apply Algorithm 2 to compute the critical points of

$$\psi(x_2, x_4) = u_1 \log |2x_2| + u_2 \log |2x_4| + u_3 \log |2x_4 - 2x_2| + u_4 \log |2x_2 + 2x_4| + v \log |x_2^2 + x_4^2 + 1|.$$

In practice, before constructing ψ , we discard redundant columns of $A|_{\mathbf{v}}$ (up to sign). Here, we keep only one among $(2, 0)$ and $(-2, 0)$, and one among the two copies of $(0, 2)$. We then solve the polynomial system

$$\begin{aligned} \frac{u_1}{x_2} - \frac{2u_3}{-2x_2 + 2x_4} + \frac{2u_4}{2x_2 + 2x_4} - \frac{2vx_2}{1 + x_2^2 + x_4^2} &= 0 \\ \frac{u_2}{x_4} + \frac{2u_3}{-2x_2 + 2x_4} + \frac{2u_4}{2x_2 + 2x_4} - \frac{2vx_4}{1 + x_2^2 + x_4^2} &= 0 \end{aligned}$$

with $(u, v) = (1, 1, 1, 1, 4)$, obtaining in about 0.06 seconds the eight points

$$(\pm 0.38268, \pm 0.92387), (\pm 0.92387, \pm 0.38268),$$

shown as green diagonal crosses in [Figure 6](#), right. In step 8, we embed these points back into $\mathbb{R}^4$. This is done by reintroducing the coordinates one at a time, in reverse order of restriction. First, the restricted v_4 (the third column of $A|_{v_1}$ in (4)) enforces $-x_3 + x_4 = 0$. Next, v_1 imposes $-x_1 - x_2 - x_3 + x_4 = 0$. Altogether, $\mathfrak{v} = \{v_1, v_4\}$ contributes to `all_points` the eight points

$$(\pm(0.38268, -0.38268), \pm(0.92387, 0.92387)), (\pm(0.92387, -0.92387), \pm(0.38268, 0.38268)).$$

This number of solutions matches one of the cases for $d = 3$ in [Table 2](#), corresponding to slices through two linearly independent vertices of C_3 . $\diamond$

Remark 3.4. Although [Algorithm 3](#) is based on numerical methods, its output can be certified from two points of view. First, `HomotopyContinuation.jl` uses interval arithmetic or Smale’s α -theory to prove that solutions of the polynomial system are real and pairwise distinct; see, e.g., [\[BBC+24\]](#) for a recent survey. Second, `CountingChambers.jl` allows exact computation of the number of maximal open cells in a hyperplane arrangement, which equals the number of solutions to the polynomial systems we consider. Together, these guarantees certify that all solutions are found, and that they are both real and distinct.

Our computations are not only certified but also numerically stable. The floating-point solutions to our systems of equations represent normals to hyperplane slices. Each such point lies in the interior of a full-dimensional cell (within an appropriate linear subspace) of the hyperplane arrangement, and within a cell the combinatorial type of the slice is constant. Thus, numerical imprecision does not affect the outcome: any small perturbation yields the same combinatorial type. Moreover, when we restrict to a lower-dimensional subspace, the resulting solutions are still floating-point approximations of normals, but the recovery of their full-dimensional coordinates is carried out by exact linear algebra. Hence, the reconstruction step introduces no additional error, and the overall computation remains numerically stable.

Finally, once we have collected one representative point from each cell of the hyperplane arrangement $\mathcal{H}_P$, we construct the corresponding slices of the polytope P and compare their combinatorial types. For this last step, we use the `Polymake` function `isomorphic` [\[GJ00\]](#).

Remark 3.5. To make the comparison efficient, we first check the f -vector of each new slice against those already obtained. If the f -vector is new, we add the slice to the list. Otherwise, we test whether the slice is isomorphic to any of the existing ones with the same f -vector. The command `isomorphic` performs this test by reducing it to checking graph isomorphism of the vertex-facet incidence graphs of the two slices.

We summarize the discussion of this section in [Algorithm 4](#). In the step where one needs to collect one representative point per cell of the hyperplane arrangement, there is the option to either use the exact ([Algorithm 1](#)) or the certified numerical ([Algorithm 3](#)). In order to stress that the numerical algorithm gives correct, certified output, we compare the two options where possible (see [Tables 3](#) and [4](#) below).

Algorithm 4 all combinatorial types of slices

INPUT: Polytope P OUTPUT: All combinatorial types of slices of P

```
1: slices  $\leftarrow$  empty
2: all_points  $\leftarrow$  either Algorithm 1( $P$ ) or Algorithm 3( $P$ )
3: for  $u \in \text{all\_points}$  do
4:   construct  $P \cap u^\perp$ 
5:   if the  $f$ -vector of  $P \cap u^\perp$  is not among the  $f$ -vector of any  $s \in \text{slices}$  then
6:     append  $P \cap u^\perp$  to slices
7:   else the  $f$ -vector of  $P \cap u^\perp$  coincides with the  $f$ -vector of the slices in  $S \subset \text{slices}$ 
8:     if  $\text{isomorphic}(P \cap u^\perp, s) = \text{true}$  for any  $s \in S$  then
9:       move on
10:    else append  $P \cap u^\perp$  to slices
11:    end if
12:  end if
13: end for
    return slices
```

3.1 Symmetries

So far, the algorithm works for any polytope. When $P = C_d$, however, the high degree of symmetry can be exploited to significantly reduce the computational complexity. While a similar reduction is possible for any polytope with a nontrivial symmetry group, we restrict the discussion to the cube for clarity.

Given a hyperplane $H = \{x \in \mathbb{R}^d \mid \langle u, x \rangle = t\}$, the slice $C_d \cap H$ is combinatorially equivalent to $C_d \cap \pi(H)$, where $\pi(H) = \{x \in \mathbb{R}^d \mid \langle \pi(u), x \rangle = t\}$ for any signed permutation $\pi \in B_d$. This follows from the fact that B_d is the full symmetry group of the cube. Hence, to accelerate the computation it is natural to exploit these symmetries and sample points from fewer cells. In principle, this can be incorporated directly into the certified numerical algorithm, Algorithm 3. Indeed, the command `HomotopyContinuation.solve` accepts an optional group input describing symmetries of the solution set. However, there is a subtle mismatch between geometry and computation. Once we add the auxiliary function g in the definition of ψ in (2), we may break some of the symmetries of the solution set. For the cube, this issue can be avoided by choosing

$$g(x) = \sum_i x_i^2 + 1,$$

which is generic enough for our purposes and invariant under the action of B_d . A priori, one could attempt a similar approach when restricting to lower-dimensional cells of the arrangement. In that case, however, the relevant symmetry group changes, and it would be necessary to select a different function g that is both generic and symmetry-preserving. Since this lies beyond the scope of the present work, in our computations we simply use the involution $x \mapsto -x$ together with the above choice of g , which remain valid under restriction to any $\mathfrak{v} \subset \text{vert}(C_d)$.

Another optional argument of `HomotopyContinuation.solve` that speeds up the computation is the “target solutions count”. This option provides a stopping criterion by specifying

the exact number of solutions that need to be found. In our setting, this information is given in [Table 2](#), whose entries were obtained using `CountingChambers.jl`.

A further way to exploit cube symmetries is in the reduction of subsets $\mathbf{v} \subset \text{vert}(C_d)$. Indeed, if $\mathbf{v} = \pi(\mathbf{v}')$ for some signed permutation $\pi \in B_d$, then the combinatorial types of the slices corresponding to hyperplanes through $\mathbf{v}$ coincide with those for hyperplanes through $\mathbf{v}'$. Here if $\mathbf{v} = \{v_1, \dots, v_k\}$, we write $\pi(\mathbf{v}) = \{\pi(v_1), \dots, \pi(v_k)\}$. The proof of this claim is given in the first paragraph of the proof of [Theorem 2.2](#). Therefore, in step 5 of [Algorithm 3](#), instead of iterating over all subsets of vertices of C_d , we may restrict to one representative tuple in each orbit

$$B_d \cdot \mathbf{v} = \{\pi(\mathbf{v}) \mid \pi \in B_d\},$$

for sets $\mathbf{v}$ consisting of at most d linearly independent vertices. With all these precautions in place, we are now ready to establish our main theorems.

Proof of [Theorem 1.1](#). Applying [Algorithm 4](#) to the cube $C_d = [-1, 1]^d \times \{1\} \subset \mathbb{R}^{d+1}$ for $d = 3, 4, 5, 6$, and using the symmetry reductions from [Section 3.1](#), we obtain exactly 4, 30, 344, and 7346 distinct combinatorial types of affine slices, respectively. $\square$

Proof of [Theorem 1.2](#). Applying [Algorithm 4](#) to the cube $C_d = [-1, 1]^d \subset \mathbb{R}^d$ for $d = 3, 4, 5, 6, 7$, again with the symmetry reductions from [Section 3.1](#), yields precisely 2, 6, 23, 133, and 1657 combinatorial types of central slices, respectively. $\square$

3.2 Refined data

In this section we highlight some further details of the slices already described in [Section 2](#), focusing on how the computation was carried out in practice. When implementing [Algorithm 4](#), we examine the cells of $\mathcal{H}_P$ dimension by dimension. Concretely, we begin with the top-dimensional cells of $\mathcal{H}_P$, which correspond to slices that do not intersect any vertex of C_d . Next, we consider cells of codimension one, which correspond to slices through exactly one vertex of C_d . After k iterations, we are examining cells of codimension k , that is, slices through k linearly independent vertices of C_d . This process continues until $k = d$. At each step, we compare the resulting set of combinatorial types with those found in the previous steps, and record how many genuinely new types arise when the slice is required to pass through one additional vertex of the cube.

We collect all this information in [Table 3](#), which reports the results for cubes up to dimension $d = 6$. An analogous summary for central slices is given in [Table 4](#). In each table, the numbers in brackets in the column indexed by k denote the combinatorial types that arise for the first time when the slice passes through k linearly independent vertices, that is, types that cannot already be obtained from slices through fewer than k vertices.

Thanks to our computation, we observe a very special behavior of central slices of the cube, which leads us to the following conjecture, verified computationally up to dimension $d = 7$.

Conjecture 4. *Let $C_d \subset \mathbb{R}^d$ be the d -dimensional cube. For each $k \in [0, d]$, let S_k denote the set of all combinatorial types of central slices of C_d through exactly k linearly independent vertices of the cube. Then, for all integers $k_1, k_2 \in [0, d]$ the intersection $S_{k_1} \cap S_{k_2}$ consists of exactly one combinatorial type, namely the type of C_{d-1} .*

	0	1	2	3	4	5	6	tot
$d = 3$	3	3 (1)	2 (0)	2 (0)				4
$d = 4$	12	14 (7)	14 (6)	10 (4)	6 (1)			30
$d = 5$	58	103 (81)	129 (96)	105 (73)	52 (31)	14 (5)		344
$d = 6$	554	1482 (1376)	2296 (2078)	2179 (1917)	1276 (1066)	422 (319)	62 (36)	7346

Table 3: Number of combinatorial types of slices of C_d in dimensions $d = 3, 4, 5, 6$. Slices are subdivided in distinct columns depending on how many linearly independent vertices of C_d they contain. Green values require the use of the numerical one. In all other cases, the exact and certified numerical outputs coincide. Numbers in brackets stand for *new* combinatorial types, that appear for the first time in that given column.

	0	1	2	3	4	5	6	tot
$d = 3$	2	1 (0)	1 (0)					2
$d = 4$	3	2 (1)	2 (1)	2 (1)				6
$d = 5$	7	6 (5)	6 (5)	5 (4)	3 (2)			23
$d = 6$	21	28 (27)	34 (33)	30 (29)	18 (17)	7 (6)		133
$d = 7$	135	288 (287)	427 (426)	419 (418)	268 (267)	105 (104)	21 (20)	1657

Table 4: Number of combinatorial types of *central* slices of the cubes in dimensions $d = 3, 4, 5, 6, 7$. We refer to Table 3 for the meaning of columns, color-coding and numbers in brackets.

We stress that in the definition of S_k we allow a slice to contain more than k vertices of C_d , but these vertices must span, together with the origin, a k -dimensional linear space. The following result shows that the intersection $S_{k_1} \cap S_{k_2}$ considered in Conjecture 4 always contains at least the combinatorial type of C_{d-1} . What remains open – and would complete the proof of the conjecture – is to show that no other combinatorial type can appear in two distinct sets S_{k_i} .

Proposition 3.6. *Let $C_d \subset \mathbb{R}^d$ be the d -dimensional cube. For every $k \in [0, d]$ there exists a central slice of C_d through exactly k linearly independent vertices of the cube which is combinatorially equivalent to C_{d-1} .*

Proof. If $k = 0$, any slice of the form $\{x_i = 0\}$ is a cube of dimension $d - 1$. If $k = d$, then any slice of the form $\{x_i = \pm x_j\}$ is again a cube of dimension $d - 1$. Now suppose $1 \leq k \leq d - 1$ and consider the vector

$$u_k = (-(d - k), \underbrace{1, \dots, 1}_{d-k}, \underbrace{0, \dots, 0}_{k-1}).$$

The hyperplane $u_k^\perp$ intersects C_d in exactly k linearly independent vertices, namely the vertices of the $(k - 1)$ -dimensional faces defined by $\{x_1 = \dots = x_{d-k+1} = \pm 1\}$. Moreover, $u_k^\perp$ intersects precisely those edges of the form $(\lambda, \pm 1, \dots, \pm 1)$ in exactly one value of λ . All remaining edges are either fully contained in $u_k^\perp$ or do not intersect it at all. Define the projection $\varphi : \mathbb{R}^d \rightarrow \mathbb{R}^{d-1}$ which forgets the first coordinate. Then, since $u_k \neq (\pm 1, 0, \dots, 0)$, the map $\varphi|_{u_k^\perp}$ is an affine isomorphism, which restricts to an affine isomorphism on $C_d \cap u_k^\perp$, whose image is C_{d-1} . Therefore, their face lattices are isomorphic, hence the slice $C_d \cap u_k^\perp$ is combinatorially equivalent to C_{d-1} . $\square$

0	1	2	3	4
[4, 6, 4]	[4, 6, 4]	[4, 6, 4]	[4, 6, 4]	[4, 6, 4]
[6, 9, 5]	[6, 9, 5]	[6, 9, 5]	[6, 9, 5]	[6, 9, 5]
[8, 12, 6]	[7, 11, 6]	[7, 11, 6]	[7, 11, 6]	[6, 12, 8]
[8, 12, 6]	[8, 12, 6]	[8, 12, 6]	[7, 12, 7]	[7, 12, 7]
[10, 15, 7]	[8, 12, 6]	[8, 12, 6]	[8, 12, 6]	[8, 12, 6]
[10, 15, 7]	[9, 14, 7]	[8, 13, 7]	[8, 13, 7]	
[10, 15, 7]	[9, 14, 7]	[8, 13, 7]	[8, 13, 7]	
[12, 18, 8]	[10, 15, 7]	[9, 14, 7]	[8, 14, 8]	
[12, 18, 8]	[10, 15, 7]	[9, 14, 7]	[9, 15, 8]	
[12, 18, 8]	[10, 15, 7]	[10, 15, 7]	[9, 15, 8]	
[12, 18, 8]	[11, 17, 8]	[10, 16, 8]		
[12, 18, 8]	[11, 17, 8]	[10, 16, 8]		
	[11, 17, 8]	[10, 16, 8]		
	[11, 17, 8]	[10, 16, 8]		

Table 5: f -vectors of all possible combinatorial types of slices of C_4 . Slices are subdivided in distinct columns depending on how many linearly independent vertices of C_4 they contain. Slices with the same combinatorial type appear in more than one column. Blue f -vectors denote combinatorial types realizable also by central slices.

The data we collected contains much more than just the number of combinatorial types of slices, but the full combinatorial data of all types of slices. For instance, one may also be interested in the f -vectors (or equivalently, the h -vectors) that can arise from slices. In particular, an open problem posed in [Kho06] asks to determine the possible h -vectors of generic affine slices of a polytope P , given the h -vector of P . Table 5 displays the f -vectors of all types of slices of the 4-dimensional cube, grouped according to the number of linearly independent vertices of C_4 that the slice contains.

4 Color classes of graphs of slices

In [NS80; FMGN97] it was computed that C_d has 12, 61 and 484 combinatorial types of slices for $d = 3, 4, 5$, respectively. The attentive reader will notice that these numbers differ from those in Theorem 1.1. Indeed, the notion of combinatorial types used in [NS80; FMGN97] differs from the standard definition [Zie95; Grü03]. In Section 4.1, we explain describe this alternative notion, which we refer to as *color class*, and provide a method to compute it. With this terminology we reproduce the computational results of [NS80; FMGN97] on the numbers of color classes of graphs of slices of C_d in dimensions $d = 3, 4, 5$. As in the previous section, these computations lead to further observations, questions, and conjectures about graphs of slices of the cube, which we present in Section 4.2.

4.1 Computation and Algorithm

Let P be a polytope and H a hyperplane. Recall that every vertex of a slice $Q = P \cap H$ is either a vertex of P or the intersection of H with the relative interior of an edge of P . We consider the vertex-edge graph $G(Q) = (V(Q), E(Q))$, where $V(Q) = \text{vert}(Q)$ and $E(Q)$ are the vertices and edges of the polytope Q , equipped with a vertex coloring

$$c : V(Q) \rightarrow \{\text{black}, \text{white}\}$$

$$v \mapsto \begin{cases} \text{white} & \text{if } v \in \text{vert}(P) , \\ \text{black} & \text{otherwise} . \end{cases}$$

A *color-preserving graph isomorphism* between two such colored graphs $G(Q_1) = (V(Q_1), E(Q_1), c_1)$ and $G(Q_2) = (V(Q_2), E(Q_2), c_2)$ and is a bijection $\varphi : V(Q_1) \rightarrow V(Q_2)$ such that

- (i) φ is a graph isomorphism, i.e., $(v, w) \in E(Q_1)$ if and only if $(\varphi(v), \varphi(w)) \in E(Q_2)$, and
- (ii) φ preserves colors, i.e., $c_1(v) = c_2(\varphi(v))$ for all $v \in V(Q_1)$.

In this language, the results of [NS80; FMGN97] count the number of colored graphs of slices of the cube, up to color-preserving graph isomorphism. We call each such isomorphism class a *color class*. Figure 7 illustrates all color classes of slices of C_3 .

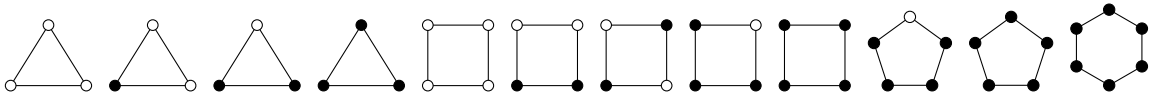


Figure 7: The 12 color classes of graphs of slices of C_3 .

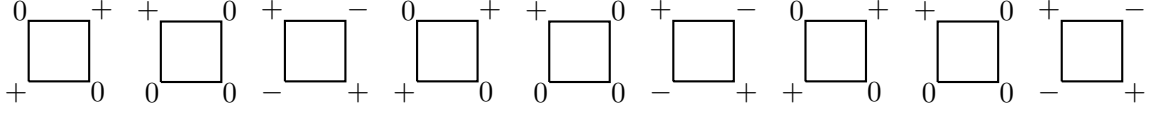


Figure 8: The 9 forbidden squares from [FMGN97].

We have verified computationally that the numbers reported in [NS80; FMGN97] are indeed the correct counts of such isomorphism classes for $d = 3, 4, 5$. In our computations, we largely follow their general approach, based on the observation that any affine hyperplane $H = \{x \in \mathbb{R}^d \mid \langle u, x \rangle + a = 0\}$ induces a vertex labeling ℓ of the vertex-edge graph of P with labels $\{+, 0, -\}$ via $\ell(v) = \text{sgn}(\langle u, v \rangle + a)$. Our goal is to determine all vertex labelings $\ell : V(P) \rightarrow \{-, 0, +\}$ that are *geometrically realizable*, i.e., for which there exists a hyperplane $H = \{x \in \mathbb{R}^d \mid \langle x, u \rangle + a = 0\}$ satisfying $\ell(v) = \text{sgn}(\langle u, v \rangle + a)$ for every $v \in V(P)$. To this end we collect four necessary conditions for a labeling ℓ to be geometrically realizable.

First, convexity of P implies that for every labeling the induced subgraph S generated by the vertices with label $+$ is connected, and the same holds for the induced subgraph of vertices labeled $-$. As the slice is unchanged when interchanging the labels $+$ and $-$, we may assume $|V(S)| \leq \frac{1}{2}|V(P)|$. Second, every vertex with label 0 is adjacent to at least one vertex with label $+$, unless the hyperplane supports a face of P (in which case no vertex is labeled $+$). In other words, every vertex with label 0 lies in the neighborhood

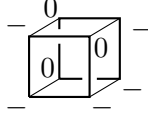
$$N(S) = \{v \in V(P) \setminus V(S) \mid \exists w \in V(S) : (v, w) \in E(P)\}.$$

Third, the list of *forbidden squares* in Figure 8 specifies all labelings of the 4-cycles in the cube graph $G(C_d)$ that are not geometrically realizable. It is straightforward to verify that this list is complete: every other square labeling occurs in some geometrically realizable labeling.

These necessary conditions are essentially those described in [FMGN97]. However, not every labeling that satisfies them is geometrically realizable. For a given labeling ℓ , the set of hyperplanes realizing it forms a relatively open polyhedral cone, defined by one equation for each vertex labeled 0 and by one strict inequality for each remaining vertex. Geometric realizability can therefore be tested by computing the closed cone $C(\ell)$ defined by non-strict inequalities (and equations), and comparing its dimension with the expected one, yielding the fourth condition. With this observation we obtain the following sketch of an algorithm for checking realizability. We point out that we do not know whether, or in what way, a check for this fourth condition was performed in the computations reported in [FMGN97], but it proved necessary in our computation in order to obtain correct labellings (see Theorem 4.1).

In the actual implementation of Algorithm 5, carried out in SageMath 10.5 [Sag24], we include checks to avoid repeating computations on labelled graphs that are identical up to labelled graph isomorphism, taking advantage of the high symmetry of the cube. The 12 color classes of graphs of slices of C_3 are depicted in Figure 7. The 61 and 484 classes for C_4 and C_5 , respectively, are printed in [FMGN97]. For detailed inspection of these $12 + 61 + 484$ classes, we provide the complete data set of color classes as raw data in our repository.

Example 4.1 (Necessary conditions are not sufficient). Consider the cube C_3 labeled as



This labeling satisfies all three necessary conditions from [FMGN97]: the induced subgraphs of vertices with labels $+$ and $-$ are connected; every vertex labeled 0 is adjacent to a vertex labeled $+$; no forbidden square occurs. Nevertheless, the unique hyperplane containing the vertices with label 0 necessarily passes through four vertices of the cube, so this labeling is not geometrically realizable. $\diamond$

Algorithm 5 Graph algorithm

INPUT: Polytope $C_d = [-1, 1]^d$

OUTPUT: List of colored vertex-edge graphs of slices of C_d

```

1: Compute vertex-edge graph  $G(C_d) = (V(C_d), E(C_d))$ 
2: color_classes  $\leftarrow$  empty
3: for connected induced subgraphs  $S$  of  $G(C_d)$  of size  $|V(S)| \leq \frac{1}{2}|V(C_d)|$  do
4:   compute the neighborhood  $N(S)$ 
5:   assign label  $+$  to all vertices in  $S$ 
6:   assign label  $-$  to all vertices in  $V(C_d) \setminus \{V(S) \cup N(S)\}$ 
7:   for  $\{-, 0\}$ -labelings of  $N(S)$  do
8:     if the labeling  $\ell$  of  $V(G)$  does not contain a forbidden square then
9:       compute the cone  $C(\ell)$ 
10:      if  $C(\ell)$  has relative interior of the expected dimension then
11:        sample a hyperplane  $H$  from the relative interior of the cone
12:        compute the graph  $K$  of  $[-1, 1]^d \cap H$ 
13:        if  $K$  not in color_classes (up to colored graph isomorphism) then
14:          store the colored graph of the slice
15:        end if
16:      end if
17:    end if
18:  end for
19: end for

```

4.2 Graphs of slices of the cube

In dimensions 4 and higher, polytopes of distinct combinatorial types (in the sense of isomorphism classes of face-lattices) can, in general, have identical vertex-edge graphs. It is therefore not expected that arbitrary slices of cubes of dimension 5 should be distinguishable by their graphs, or their respective color class. However, analyzing the computed color classes, we observe that indeed all combinatorial types are distinguished by their vertex-edge graphs, thus justifying the computational method explained in Section 4.1. While this holds true for $d \leq 5$, we do not know whether this remains true for slices of cubes in higher dimensions.

Question 1. *Do all distinct combinatorial types of slices of the cube C_d have nonisomorphic vertex-edge graphs? Equivalently, are slices of cubes reconstructible from their graphs, given the additional information that they are indeed slices of the cube?*

We now collect evidence both for and against an affirmative answer to this question. First, note that every simple polytope is reconstructible from its graph ([BML87; Kal88]). More generally, polytopes with at most two nonsimple vertices are reconstructible from their graphs [DNPV+18]. This is key to the following statement.

Proposition 4.2. *Any slice of the cube C_d which contains at most 2 vertices of C_d is reconstructible from its graph.*

Proof. Recall that a vertex of a slice $C_d \cap H$ is either a vertex of C_d or arises as the intersection of H with the relative interior of an edge e . Since any polytope with at most 2 nonsimple vertices is reconstructible, it suffices to show that a vertex of the slice $C_d \cap H$ of the form $v = \text{relint}(e) \cap H$ is simple, i.e., has degree $d - 1$.

The edges of the slice incident to v are intersections of squares of C_d containing e . Since H intersects e in its relative interior, all adjacent squares are intersected in their respective relative interiors as well. We thus need to show that e is contained in precisely $d - 1$ squares. For $d \geq 2$ we have $C_d = C_{d-1} \times C_1$, and by symmetry it suffices to check the statement for the edge $e = (-1, \dots, -1, \lambda)$, $\lambda \in [-1, 1]$. The squares of C_d containing e are precisely the sets $\tilde{e} \times C_1$, where $\tilde{e}$ is an edge of $C_{d-1} \times \{-1\}$ containing the vertex $(-1, \dots, -1) \in C_{d-1}$. As C_{d-1} is simple, the number of such edges is $d - 1$. $\square$

On the other hand, [PVS22, Example 23] provide a polytope Q whose graph is that of the hypersimplex $\Delta(2, 5)$ – itself a slice of C_5 – but Q and $\Delta(2, 5)$ are not combinatorially equivalent. The key point is that Q cannot be realized as a slice of a cube. Therefore, the additional information that the graph comes from a slice of a cube is genuinely needed to reconstruct such slices from their graphs up to dimension 5. It remains open whether this is true in all dimensions. Likewise, the more restrictive question of whether a slice is reconstructible from its *colored* graph is also unresolved in general.

Finally, although the d -dimensional cube is simple – therefore uniquely determined by its vertex-edge graph as a d -dimensional polytope – there exists, for every k with $3 \leq k < d$, a k -dimensional polytope P whose graph is isomorphic to that of the d -cube [JZ00]. More generally, for integers r, k satisfying $2r + 2 \leq k \leq d$, there exists a k -polytope whose r -skeleton is combinatorially equivalent to that of the d -cube.

Question 2. *Let P be a k -dimensional polytope whose graph (or more generally whose r -skeleton) is isomorphic to that of the d -dimensional cube. How do the graphs of $(k - 1)$ -dimensional slices of P compare with the graphs of the $(d - 1)$ -dimensional slices of the cube?*

References

- [Abe18] Ulrich Abel. “The number of gridpoints on hyperplane sections of the d -dimensional cube”. In: *Proceedings of the American Mathematical Society* 146.12 (2018), pp. 5349–5355. DOI: [10.1090/proc/14233](https://doi.org/10.1090/proc/14233).
- [Abe77] Harold Abelson. “Computational geometry of linear threshold functions”. In: *Information and Control* 34.1 (1977), pp. 66–92. DOI: [10.1016/s0019-9958\(77\)90272-8](https://doi.org/10.1016/s0019-9958(77)90272-8).

- [AG24] Gergely Ambrus and Barnabás Gárgyán. “Non-diagonal critical central sections of the cube”. In: *Advances in Mathematics* 441 (2024), p. 109524. DOI: [10.1016/j.aim.2024.109524](https://doi.org/10.1016/j.aim.2024.109524).
- [Ali08] Iskander Aliev. “Siegel’s Lemma and Sum-Distinct Sets”. In: *Twentieth Anniversary Volume*: Springer New York, 2008, pp. 1–8. DOI: [10.1007/978-0-387-87363-3_4](https://doi.org/10.1007/978-0-387-87363-3_4).
- [Ali20] Iskander Aliev. “On the volume of hyperplane sections of a d-cube”. In: *Acta Mathematica Hungarica* 163.2 (2020), pp. 547–551. DOI: [10.1007/s10474-020-01057-y](https://doi.org/10.1007/s10474-020-01057-y).
- [Amb22] Gergely Ambrus. “Critical central sections of the cube”. In: *Proceedings of the American Mathematical Society* 150.10 (2022), pp. 4463–4474. DOI: [10.1090/proc/15955](https://doi.org/10.1090/proc/15955).
- [Bal86] Keith Ball. “Cube slicing in $\mathbb{R}^n$ ”. In: *Proceedings of the American Mathematical Society* 97.3 (1986), pp. 465–473. DOI: [10.1090/s0002-9939-1986-0840631-0](https://doi.org/10.1090/s0002-9939-1986-0840631-0).
- [Bal87] Keith Ball. “Volumes of sections of cubes and related problems”. In: *Geometric Aspects of Functional Analysis*. Springer Berlin Heidelberg, 1987, pp. 251–260. DOI: [10.1007/bfb0090058](https://doi.org/10.1007/bfb0090058).
- [BBC+24] Daniel J. Bates, Paul Breiding, Tianran Chen, Jonathan D. Hauenstein, Anton Leykin, and Frank Sottile. *Numerical Nonlinear Algebra*. 2024. arXiv: [2302.08585](https://arxiv.org/abs/2302.08585).
- [BBMS22] Katalin Berlow, Marie-Charlotte Brandenburg, Chiara Meroni, and Isabelle Shankar. “Intersection bodies of polytopes”. In: *Beiträge zur Algebra und Geometrie* 63.2 (2022), pp. 419–439. DOI: [10.1007/s13366-022-00621-7](https://doi.org/10.1007/s13366-022-00621-7).
- [BDLM25] Marie-Charlotte Brandenburg, Jesús A. De Loera, and Chiara Meroni. “The best ways to slice a polytope”. In: *Mathematics of Computation* 94.352 (2025), pp. 1003–1042. DOI: [10.1090/mcom/4006](https://doi.org/10.1090/mcom/4006).
- [BEK23] Taylor Brysiewicz, Holger Eble, and Lukas Kühne. “Computing Characteristic Polynomials of Hyperplane Arrangements with Symmetries”. In: *Discrete & Computational Geometry* 70.4 (2023), pp. 1356–1377. DOI: [10.1007/s00454-023-00557-2](https://doi.org/10.1007/s00454-023-00557-2).
- [BEKS17] Jeff Bezanson, Alan Edelman, Stefan Karpinski, and Viral B Shah. “Julia: A fresh approach to numerical computing”. In: *SIAM Review* 59.1 (2017), pp. 65–98. DOI: [10.1137/141000671](https://doi.org/10.1137/141000671).
- [BF22] Imre Bárány and Péter Frankl. “Cells in the box and a hyperplane”. In: *Journal of the European Mathematical Society* 25.7 (2022), pp. 2863–2877. DOI: [10.4171/jems/1252](https://doi.org/10.4171/jems/1252).
- [BFGM20] Ferenc Ágoston Bartha, Ferenc Fodor, and Bernardo González Merino. “Central Diagonal Sections of the n-Cube”. In: *International Mathematics Research Notices* 2021.4 (2020), pp. 2861–2881. DOI: [10.1093/imrn/rnaa254](https://doi.org/10.1093/imrn/rnaa254).
- [BML87] Roswitha Blind and Peter Mani-Levitska. “Puzzles and polytope isomorphisms”. In: *Aequationes Mathematicae* 34.2–3 (1987), pp. 287–297. DOI: [10.1007/bf01830678](https://doi.org/10.1007/bf01830678).
- [BP56] Herbert Busemann and Clinton M. Petty. “Problems on convex bodies”. In: *Mathematica Scandinavica* 4 (1956), pp. 88–94. DOI: [10.7146/math.scand.a-10457](https://doi.org/10.7146/math.scand.a-10457).
- [BSW24] Paul Breiding, Bernd Sturmfels, and Kexin Wang. *Computing Arrangements of Hypersurfaces*. 2024. arXiv: [2409.09622](https://arxiv.org/abs/2409.09622).
- [BT18] Paul Breiding and Sascha Timme. “HomotopyContinuation.jl: A Package for Homotopy Continuation in Julia”. In: *Mathematical Software – ICMS 2018*. Ed. by James H. Davenport, Manuel Kauers, George Labahn, and Josef Urban. Springer International Publishing, 2018, pp. 458–465.
- [CDCBA22] Zejia Chen, Fabing Duan, François Chapeau-Blondeau, and Derek Abbott. “Training threshold neural networks by extreme learning machine and adaptive stochastic resonance”. In: *Physics Letters A* 432 (2022), p. 128008. DOI: <https://doi.org/10.1016/j.physleta.2022.128008>.

- [CL91] Don Chakerian and Dave Logothetti. “Cube Slices, Pictorial Triangles, and Probability”. In: *Mathematics Magazine* 64.4 (1991), pp. 219–241. DOI: [10.1080/0025570x.1991.11977612](https://doi.org/10.1080/0025570x.1991.11977612).
- [Cov64] Thomas M. Cover. “Geometrical and Statistical Properties of Linear Threshold Devices”. PhD thesis. 1964, p. 77. URL: <https://isl.stanford.edu/~cover/papers/paper1.pdf>.
- [DEF+25] Wolfram Decker, Christian Eder, Claus Fieker, Max Horn, and Michael Joswig, eds. *The Computer Algebra System OSCAR: Algorithms and Examples*. 1st ed. Vol. 32. Algorithms and Computation in Mathematics. Springer, 2025. DOI: [10.1007/978-3-031-62127-7](https://doi.org/10.1007/978-3-031-62127-7).
- [DLLCT25] Jesús A. De Loera, Gyivan Lopez-Campos, and Antonio J. Torres. *On the Number of Vertices in a Hyperplane Section of a Polytope*. 2025. arXiv: [2412.12419](https://arxiv.org/abs/2412.12419).
- [DNPV+18] Joseph Doolittle, Eran Nevo, Guillermo Pineda-Villavicencio, Julien Ugon, and David Yost. “On the Reconstruction of Polytopes”. In: *Discrete & Computational Geometry* 61.2 (2018), pp. 285–302. DOI: [10.1007/s00454-018-9997-9](https://doi.org/10.1007/s00454-018-9997-9).
- [DVJL24] Antony Della Vecchia, Michael Joswig, and Benjamin Lorenz. “A FAIR file format for mathematical software”. In: *Mathematical software—ICMS 2024*. Vol. 14749. Lecture Notes in Comput. Sci. Springer, Cham, 2024, pp. 234–244. DOI: [10.1007/978-3-031-64529-7_25](https://doi.org/10.1007/978-3-031-64529-7_25).
- [EGLP23] Tolga Ergen, Halil Ibrahim Gulluk, Jonathan Lacotte, and Mert Pilanci. *Globally Optimal Training of Neural Networks with Threshold Activation Functions*. 2023. arXiv: [2303.03382](https://arxiv.org/abs/2303.03382).
- [Eve10] D. Evers. “Hyperebenenschnitte des vierdimensionalen Würfels”. Wissenschaftliche Prüfungsarbeit. Universität Koblenz-Landau, Koblenz, 2010.
- [FH24] Ansgar Freyer and Martin Henk. “Polynomial bounds in Koldobsky’s discrete slicing problem”. In: *Proceedings of the American Mathematical Society* 152.7 (2024), pp. 3063–3074. DOI: [10.1090/proc/16753](https://doi.org/10.1090/proc/16753).
- [FM24] Luis Ferroni and Daniel McGinnis. “Lattice points in slices of prisms”. In: *Canadian Journal of Mathematics* 77.3 (2024), pp. 1013–1040. DOI: [10.4153/s0008414x24000233](https://doi.org/10.4153/s0008414x24000233).
- [FMGN97] Hiroshi Fukuda, Nobuaki Muto, Kikuko Goto, and Gisaku Nakamura. “Sections of hyper-cube in five dimensions”. In: *FORMA* 12 (1997), p. 15. URL: https://www.researchgate.net/publication/236966264_Sections_of_hyper-cube_in_five_dimensions.
- [FR12] Rolfdieter Frank and Harald Riede. “Hyperplane Sections of the n-Dimensional Cube”. In: *The American Mathematical Monthly* 119.10 (2012), p. 868. DOI: [10.4169/amer.math.monthly.119.10.868](https://doi.org/10.4169/amer.math.monthly.119.10.868).
- [Gar94] Richard J. Gardner. “Intersection bodies and the Busemann-Petty problem”. In: *Transactions of the American Mathematical Society* 342.1 (1994), pp. 435–445. DOI: [10.2307/2154703](https://doi.org/10.2307/2154703).
- [GJ00] Ewgenij Gawrilow and Michael Joswig. “polymake: a Framework for Analyzing Convex Polytopes”. In: *Polytopes — Combinatorics and Computation*. Ed. by Gil Kalai and Günter M. Ziegler. Basel: Birkhäuser Basel, 2000, pp. 43–73. DOI: [10.1007/978-3-0348-8438-9_2](https://doi.org/10.1007/978-3-0348-8438-9_2).
- [GKS99] Richard J. Gardner, Alexander Koldobsky, and Thomas Schlumprecht. “An analytic solution to the Busemann-Petty problem on sections of convex bodies”. In: *Annals of Mathematics. Second Series* 149.2 (1999), pp. 691–703. DOI: [10.2307/120978](https://doi.org/10.2307/120978).
- [GMP21] Samuel C. Gutekunst, Karola Mészáros, and T. Kyle Petersen. “Root Cones and the Resonance Arrangement”. In: *The Electronic Journal of Combinatorics* 28.1 (2021). DOI: [10.37236/8759](https://doi.org/10.37236/8759).

- [Grü03] Branko Grünbaum. *Convex polytopes*. 2nd ed. Vol. 221. Graduate Texts in Mathematics. Springer-Verlag, New York, 2003, pp. xvi+468. DOI: [10.1007/978-1-4613-0019-9](https://doi.org/10.1007/978-1-4613-0019-9).
- [GS93] Peter Gritzmann and Bernd Sturmfels. “Minkowski Addition of Polytopes: Computational Complexity and Applications to Gröbner Bases”. In: *SIAM Journal on Discrete Mathematics* 6.2 (1993), pp. 246–269. DOI: [10.1137/0406019](https://doi.org/10.1137/0406019).
- [Had72] Hugo Hadwiger. “Gitterperiodische Punktmengen und Isoperimetrie”. In: *Monatshefte für Mathematik* 76.5 (1972), pp. 410–418. DOI: [10.1007/bf01297304](https://doi.org/10.1007/bf01297304).
- [Hen79] Douglas Hensley. “Slicing the cube in $\mathbb{R}^n$ and probability (bounds for the measure of a central cube slice in $\mathbb{R}^n$ by probability methods)”. In: *Proceedings of the American Mathematical Society* 73.1 (1979), pp. 95–100. DOI: [10.1090/s0002-9939-1979-0512066-8](https://doi.org/10.1090/s0002-9939-1979-0512066-8).
- [Huh13] June Huh. “The maximum likelihood degree of a very affine variety”. In: *Compositio Mathematica* 149.8 (2013), pp. 1245–1266. DOI: [10.1112/S0010437X13007057](https://doi.org/10.1112/S0010437X13007057).
- [IT21] Grigory Ivanov and Igor Tsiutsiurupa. “On the Volume of Sections of the Cube”. In: *Analysis and Geometry in Metric Spaces* 9.1 (2021), pp. 1–18. DOI: [10.1515/agms-2020-0103](https://doi.org/10.1515/agms-2020-0103).
- [JZ00] Michael Joswig and Gunter M. Ziegler. “Neighborly Cubical Polytopes”. In: *Discrete & Computational Geometry* 24.2 (2000), pp. 325–344. DOI: [10.1007/s004540010039](https://doi.org/10.1007/s004540010039).
- [Kal88] Gil Kalai. “A simple way to tell a simple polytope from its graph”. In: *Journal of Combinatorial Theory, Series A* 49.2 (1988), pp. 381–383. DOI: [10.1016/0097-3165\(88\)90064-7](https://doi.org/10.1016/0097-3165(88)90064-7).
- [KCB23] Sammy Khalife, Hongyu Cheng, and Amitabh Basu. “Neural networks with linear threshold activations: structure and algorithms”. In: *Mathematical Programming* 206.1–2 (2023), pp. 333–356. DOI: [10.1007/s10107-023-02016-5](https://doi.org/10.1007/s10107-023-02016-5).
- [Kho06] Askold Khovanskii. “Combinatorics of sections of polytopes and Coxeter groups in Lobachevsky spaces”. In: *The Coxeter legacy*. Amer. Math. Soc., Providence, RI, 2006, pp. 129–157.
- [KK11] Hermann König and Alexander Koldobsky. “Volumes of low-dimensional slabs and sections in the cube”. In: *Advances in Applied Mathematics* 47.4 (2011), pp. 894–907. DOI: [10.1016/j.aam.2011.05.001](https://doi.org/10.1016/j.aam.2011.05.001).
- [KK19] Hermann König and Alexander Koldobsky. “On the maximal perimeter of sections of the cube”. In: *Advances in Mathematics* 346 (2019), pp. 773–804. DOI: [10.1016/j.aim.2019.02.017](https://doi.org/10.1016/j.aim.2019.02.017).
- [KL25] Boaz Klartag and Joseph Lehec. “Affirmative Resolution of Bourgain’s Slicing Problem Using Guan’s Bound”. In: *Geometric and Functional Analysis* 35.4 (2025), pp. 1147–1168. DOI: [10.1007/s00039-025-00718-w](https://doi.org/10.1007/s00039-025-00718-w).
- [Kol05] Alexander Koldobsky. *Fourier analysis in convex geometry*. Vol. 116. Mathematical Surveys and Monographs. American Mathematical Society, Providence, RI, 2005, pp. vi+170. DOI: [10.1090/surv/116](https://doi.org/10.1090/surv/116).
- [Kol98] Alexander Koldobsky. “Intersection bodies in $\mathbb{R}^4$ ”. In: *Advances in Mathematics* 136.1 (1998), pp. 1–14. DOI: [10.1006/aima.1998.1718](https://doi.org/10.1006/aima.1998.1718).
- [Kön21] Hermann König. “Non-central sections of the simplex, the cross-polytope and the cube”. In: *Advances in Mathematics* 376 (2021), p. 107458. DOI: [10.1016/j.aim.2020.107458](https://doi.org/10.1016/j.aim.2020.107458).
- [Kön25] Hermann König. “On maximal hyperplane sections of the unit ball of l_p^n for $p > 2$ ”. In: *Advances in Operator Theory* 10.1 (2025), Paper No. 15, 42. DOI: [10.1007/s43036-024-00404-y](https://doi.org/10.1007/s43036-024-00404-y).

- [KP20] Lars Kastner and Marta Panizzut. “Hyperplane arrangements in polymake”. In: *Mathematical software—ICMS 2020*. Vol. 12097. Lecture Notes in Comput. Sci. Springer, Cham, 2020, pp. 232–240. DOI: [10.1007/978-3-030-52200-1_23](https://doi.org/10.1007/978-3-030-52200-1_23).
- [Lon00] Yossi Lonke. “On Random Sections of the Cube”. In: *Discrete & Computational Geometry* 23.2 (2000), pp. 157–169. DOI: [10.1007/p100009493](https://doi.org/10.1007/p100009493).
- [Lut88] Erwin Lutwak. “Intersection bodies and dual mixed volumes”. In: *Advances in Mathematics* 71.2 (1988), pp. 232–261. DOI: [10.1016/0001-8708\(88\)90077-1](https://doi.org/10.1016/0001-8708(88)90077-1).
- [MM08] Jean-Luc Marichal and Michael J. Mossinghoff. “Slices, Slabs, and Sections of the Unit Hypercube”. In: *Open Journal of Analytic Combinatorics* 3 (2008), pp. 1–23. DOI: [10.61091/ojac-301](https://doi.org/10.61091/ojac-301).
- [MM15] Guido F. Montúfar and Jason Morton. “When Does a Mixture of Products Contain a Product of Mixtures?” In: *SIAM Journal on Discrete Mathematics* 29.1 (2015), pp. 321–347. DOI: [10.1137/140957081](https://doi.org/10.1137/140957081).
- [MP88] Mathieu Meyer and Alain Pajor. “Sections of the unit ball of L_p^n ”. In: *Journal of Functional Analysis* 80.1 (1988), pp. 109–123. DOI: [10.1016/0022-1236\(88\)90068-7](https://doi.org/10.1016/0022-1236(88)90068-7).
- [MSZZ13] James Moody, Corey Stone, David Zach, and Artem Zvavitch. “A Remark on the Extremal Non-Central Sections of the Unit Cube”. In: *Asymptotic Geometric Analysis*. Springer New York, 2013, pp. 211–228. DOI: [10.1007/978-1-4614-6406-8_9](https://doi.org/10.1007/978-1-4614-6406-8_9).
- [MTB70] Saburo Muroga, Takashi Tsuboi, and Charles R. Baugh. “Enumeration of Threshold Functions of Eight Variables”. In: *IEEE Transactions on Computers* C-19.9 (1970), pp. 818–825. DOI: [10.1109/t-c.1970.223046](https://doi.org/10.1109/t-c.1970.223046).
- [Mur62] Saburo Muroga. “Generation of self-dual threshold functions and lower bounds of the number of threshold functions and a maximum weight”. In: *3rd Annual Symposium on Switching Circuit Theory and Logical Design (SWCT 1962)*. IEEE, 1962, pp. 169–184. DOI: [10.1109/focs.1962.7](https://doi.org/10.1109/focs.1962.7).
- [NS80] Gisaku Nakamura and Akio Sasaki. “Puzzle V”. Japanese. In: *Bessatu Suri-kagaku* 10 (1980), pp. 36–42.
- [NT23] Piotr Nayar and Tomasz Tkocz. “Extremal sections and projections of certain convex bodies: a survey”. In: *Harmonic Analysis and Convexity*. De Gruyter, 2023, pp. 343–390. DOI: [10.1515/9783110775389-008](https://doi.org/10.1515/9783110775389-008).
- [Ojh00] Piyush C. Ojha. “Enumeration of linear threshold functions from the lattice of hyperplane intersections”. In: *IEEE Transactions on Neural Networks* 11.4 (2000), pp. 839–850. DOI: [10.1109/72.857765](https://doi.org/10.1109/72.857765).
- [Ole03] Krzysztof Oleszkiewicz. “On p -pseudostable random variables, Rosenthal spaces and l_p^n ball slicing”. In: *Geometric aspects of functional analysis*. Vol. 1807. Lecture Notes in Math. Springer, Berlin, 2003, pp. 188–210. DOI: [10.1007/978-3-540-36428-3_16](https://doi.org/10.1007/978-3-540-36428-3_16).
- [O’N71] Patrick E. O’Neil. “Hyperplane cuts of an n -cube”. In: *Discrete Mathematics* 1.2 (1971), pp. 193–195. DOI: [10.1016/0012-365x\(71\)90025-2](https://doi.org/10.1016/0012-365x(71)90025-2).
- [Osc] OSCAR – Open Source Computer Algebra Research system, Version 1.4.1. The OSCAR Team, 2025. URL: <https://www.oscar-system.org>.
- [Pou23a] Lionel Pournin. “Local extrema for hypercube sections”. In: *Journal d’Analyse Mathématique* 152.2 (2023), pp. 557–594. DOI: [10.1007/s11854-023-0304-1](https://doi.org/10.1007/s11854-023-0304-1).
- [Pou23b] Lionel Pournin. “Shallow Sections of the Hypercube”. In: *Israel Journal of Mathematics* 255.(2) (2023), pp. 685–704. DOI: [10.1007/s11856-022-2400-9](https://doi.org/10.1007/s11856-022-2400-9).
- [Pou25] Lionel Pournin. *Deep sections of the hypercube*. 2025. arXiv: [2407.04637](https://arxiv.org/abs/2407.04637).
- [PVS22] Guillermo Pineda-Villavicencio and Benjamin Schröter. “Reconstructibility of Matroid Polytopes”. In: *SIAM Journal on Discrete Mathematics* 36.1 (2022), pp. 490–508. DOI: [10.1137/21m1401176](https://doi.org/10.1137/21m1401176).

- [RW24] Bernhard Reinke and Kexin Wang. *Hypersurface Arrangements with Generic Hypersurfaces Added*. 2024. arXiv: [2412.20869](https://arxiv.org/abs/2412.20869).
- [Sag24] The Sage Developers. *SageMath, the Sage Mathematics Software System (Version 10.5)*. 2024. URL: <https://www.sagemath.org>.
- [Sak93] Michael E. Saks. “Slicing the hypercube”. In: *Surveys in Combinatorics, 1993*. Cambridge University Press, 1993, pp. 211–256. DOI: [10.1017/cbo9780511662089.009](https://doi.org/10.1017/cbo9780511662089.009).
- [Swa16] Hunter Swan. “Expected Number of Vertices of a Hypercube Slice”. In: *The American Mathematical Monthly* 123.6 (2016), p. 600. DOI: [10.4169/amer.math.monthly.123.6.600](https://doi.org/10.4169/amer.math.monthly.123.6.600).
- [Vaa79] Jeffrey Vaaler. “A geometric inequality with applications to linear forms”. In: *Pacific Journal of Mathematics* 83.2 (1979), pp. 543–553. DOI: [10.2140/pjm.1979.83.543](https://doi.org/10.2140/pjm.1979.83.543).
- [Var95] Alexander Varchenko. “Critical points of the product of powers of linear functions and families of bases of singular vectors”. In: *Compositio Mathematica* 97.3 (1995), pp. 385–401. URL: http://www.numdam.org/item?id=CM_1995__97_3_385_0.
- [WAP00] Walter Wenzel, Nihat Ay, and Frank Pasemann. “Hyperplane Arrangements Separating Arbitrary Vertex Classes in n-Cubes”. In: *Advances in Applied Mathematics* 25.3 (2000), pp. 284–306. DOI: [10.1006/aama.2000.0701](https://doi.org/10.1006/aama.2000.0701).
- [Win65] Robert O. Winder. “Enumeration of Seven-Argument Threshold Functions”. In: *IEEE Transactions on Electronic Computers* EC-14.3 (1965), pp. 315–325. DOI: [10.1109/pegc.1965.264136](https://doi.org/10.1109/pegc.1965.264136).
- [Zie95] Günter M. Ziegler. *Lectures on polytopes*. Vol. 152. Graduate Texts in Mathematics. Springer-Verlag, New York, 1995, pp. x+370. DOI: [10.1007/978-1-4613-8431-1](https://doi.org/10.1007/978-1-4613-8431-1).
- [Zon05] Chuanming Zong. “What is known about unit cubes”. In: *Bulletin of the American Mathematical Society* 42.02 (2005), pp. 181–212. DOI: [10.1090/s0273-0979-05-01050-5](https://doi.org/10.1090/s0273-0979-05-01050-5).
- [Zon06] Chuanming Zong. *The Cube-A Window to Convex and Discrete Geometry*. Cambridge University Press, 2006. DOI: [10.1017/cbo9780511543173](https://doi.org/10.1017/cbo9780511543173).
- [Zue92] Yu. A. Zuev. “Methods of geometry and probabilistic combinatorics in threshold logic”. In: *Discrete Mathematics and Applications* 2.4 (1992). DOI: [10.1515/dma.1992.2.4.427](https://doi.org/10.1515/dma.1992.2.4.427).
- [Zva08] Artem Zvavitch. “Gaussian measure of sections of dilates and translations of convex bodies”. In: *Advances in Applied Mathematics* 41.2 (2008), pp. 247–254. DOI: [10.1016/j.aam.2007.03.003](https://doi.org/10.1016/j.aam.2007.03.003).

MARIE-CHARLOTTE BRANDENBURG
 RUHR-UNIVERSITÄT BOCHUM
marie-charlotte.brandenburg@rub.de

CHIARA MERONI
 ETH INSTITUTE FOR THEORETICAL STUDIES
chiara.meroni@eth-its.ethz.ch

Appendix A Distribution of slices

We show the distribution of the combinatorial types of slices according to the number of vertices of the slice. For the cubes C_5 (see Figure 9) and C_6 (see Figure 10), we display several classes of slices: all affine slices (light green), affine generic slices (dark green), all central slices (light blue), and generic central slices (dark blue). This analysis for the cube C_4 was already shown in Figure 4, with the same color-coding. For the cube C_7 , we display all central slices (light blue) and generic central slices (dark blue) in Figure 11.

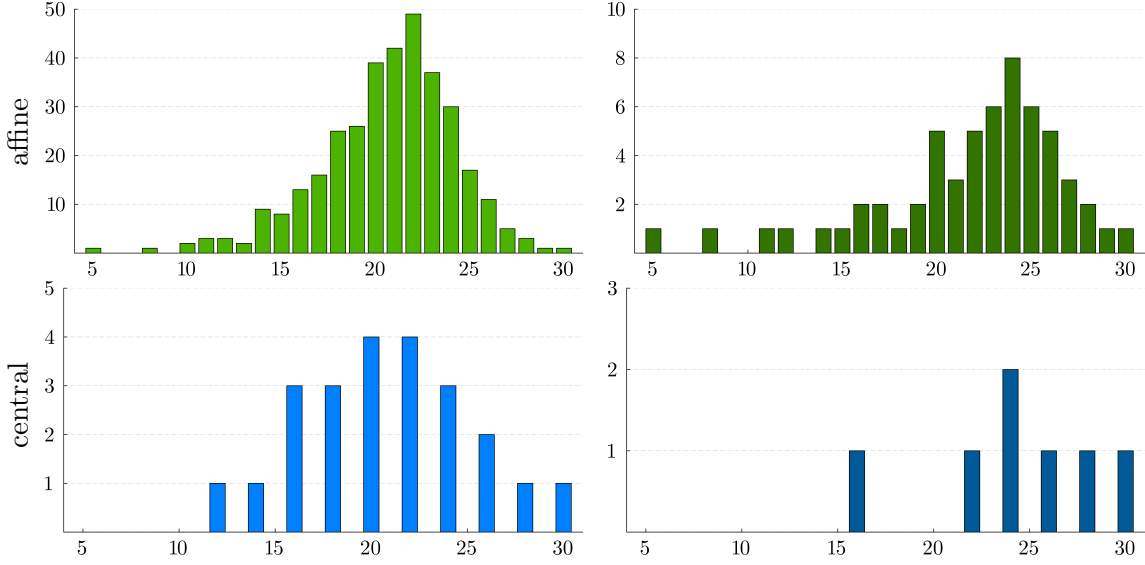


Figure 9: Number of combinatorial types of slices of C_5 by number of vertices. Notice that the vertical axis has a different scaling in each plot.

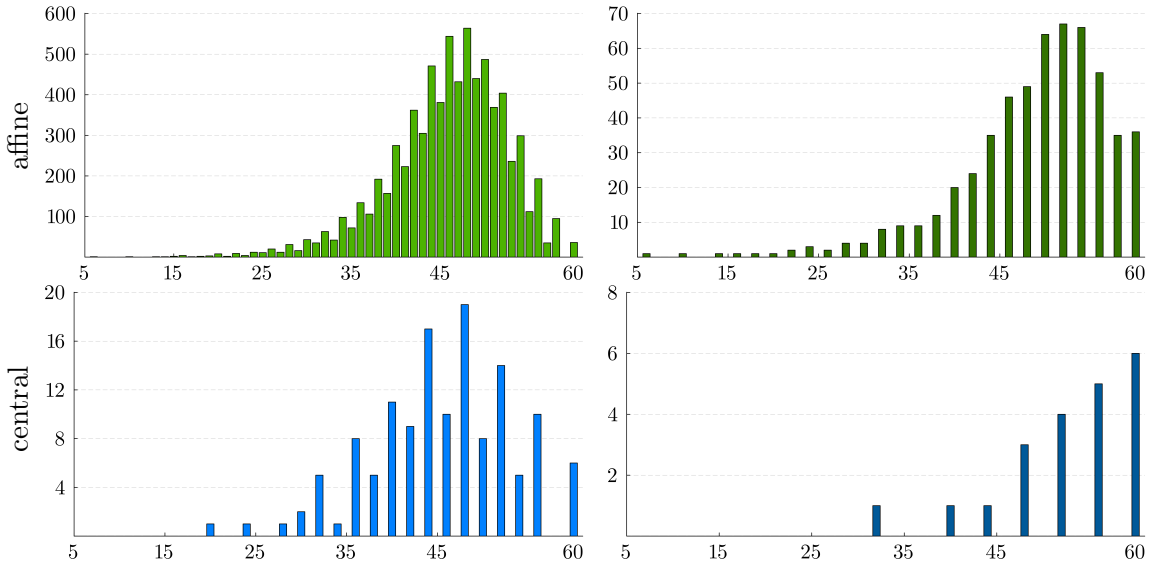


Figure 10: Number of combinatorial types of slices of C_6 by number of vertices. Notice that the vertical axis has a different scaling in each plot.

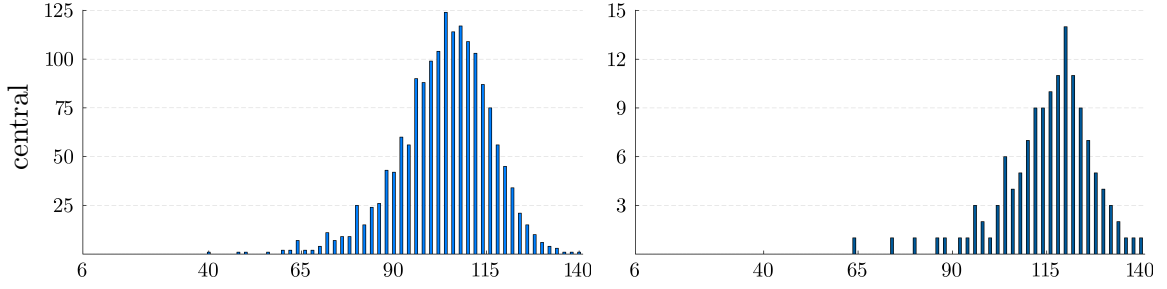


Figure 11: Number of combinatorial types of central slices of C_7 by number of vertices. Notice that the vertical axis has a different scaling in each plot.

Appendix B Repository

All data produced and used in our computations are collected in a public repository

<https://doi.org/10.5281/zenodo.17304584>

This appendix is intended to serve as a guide to the repository so that readers can locate and interpret all the data generated in our computations.

The repository provides both intermediate and final output data related to the computation of slices of cubes C_d . To guarantee reproducibility, we include the Julia environment files `Project.toml` and `Manifest.toml`, which specify the exact Julia version and package dependencies used in the notebook `NAGSlicesCubes.ipynb`. This notebook provides the computational workflow for Algorithm 4, using the certified numerical algorithm Algorithm 3. The main functions are collected in `Functions.jl`. The implementation of the exact Algorithm 1 is already available at <https://mathrepo.mis.mpg.de/BestSlicePolytopes>.

The bulk of the repository consists of the results of the computations discussed in Section 3. Several folders record data for each cube dimension of $d = 4, 5, 6, 7$, with a dedicated subfolder for every dimension. Two sets of data correspond to different stages of the computation.

Preliminary computations. Intermediate steps of the algorithm are stored in the folder ‘Data-VertexTuples’, which contains two types of files. The file `dcubeVertexTuples.txt` lists all vertex tuples $\mathbf{v}$ of C_d (up to the action of the symmetry group B_d) that are needed to restrict the initial hyperplane arrangement, in step 5 of Algorithm 3. The computation of these tuples is discussed in Section 3.1. In each file, the i -th line corresponds to tuples of $i + 1$ linearly independent vertices (in the case of one vertex, there is a unique candidate up to B_d , so any choice of vertex would do the job). The file `dcubeCert.txt` records, for each such tuple $\mathbf{v}$, the number of cells in the restricted hyperplane arrangement $\mathcal{H}_{A|_{\mathbf{v}}}$, using the same line-by-line convention. These are the numbers stored in Table 2.

Final slice data. The final output of Algorithm 4 is contained in the folder ‘Data-Slices’, which is organized into two subfolders: ‘JSON’, containing the files in `.mrdf` format [DVJL24], ‘TXT’, containing the files in `.txt` format. While the `.txt` format is easily readable, it contains numerical imprecision. On the other hand, the `.mrdf` format can be interpreted by default only with Julia (see [DVJL24, Section 3] for usage in other software systems), and produces numerical polytopes in OSCAR without numerical inconsistencies.

Each of these two folders contains four subfolders named ‘4cube’, ‘5cube’, ‘6cube’, and ‘7cube’, corresponding to the dimension of the ambient cube. Within each of these subfolders, data files are organized according to the following naming scheme. Files ending in `@int` list all combinatorial types of generic slices (those containing no vertices of C_d), files ending in `@nv` list all combinatorial types of slices passing through n linearly independent vertices, and files ending in `@tot` give the complete set of combinatorial types. The symbol `@` encodes the type of data: `f` for f -vectors, `u` for slice normals, and `s` for slice vertices. When an additional `c` follows `@` (for example `fcint`), the file contains only central slices, i.e., slices passing through the origin.

Graph data. A further component of the repository records the slices in terms of their colored vertex-edge graphs, up to color class, as discussed in [Section 4](#). The folder ‘Graphs’ contains, for each dimension $d = 3, 4, 5$, a file `dcubeGraphVerts.txt` obtained as output of [Algorithm 5](#). Every line of such a file lists the vertices of one slice of the d -dimensional cube, taken up to colored graph isomorphism. From these vertex sets one can reconstruct the colored graph of each slice by computing the convex hull to obtain the corresponding polytope, extracting its graph, and finally assigning colors to the vertices: a vertex is colored white if all its coordinates are either $+1$ or -1 , and black otherwise.