On the Number of Compositions of Two Polycubes*

Andrei Asinowski[†] Gill Barequet[‡] Gil Ben-Shachar[§]
Martha Carolina Osegueda[¶] Günter Rote[‖]

July 3, 2023

Abstract

A composition of two polycubes is appending them to each other so that the union is a valid polycube. We provide almost tight (up to subpolynomial factors) bounds on the minimum and maximum possible numbers of compositions of two polycubes, either when each is of size n, or when their total size is $2n$, in two and higher dimensions. We also provide an efficient algorithm (with some trade-off between time and space) for computing the number of compositions that two given polyominoes (or polycubes) have.

Keywords: Polyominoes, polycubes.

1 Introduction

A d-dimensional *polycube* (*polyomino* if $d = 2$) is a connected set of cells on the cubical lattice $\mathbb{Z}^d$, where the connectivity is through $(d-1)$-dimensional faces. Polycubes and other *lattice animals* (e.g., polyiamonds and polyhexes) play for more than half a century an important role in enumerative combinatorics [5] as well as in statistical physics [4].

The *size* (volume, or area in the plane) of a polycube is the number of d-dimensional cells it contains. A *composition* of two d-dimensional polycubes is the placement of one of them relative to the other, such that they touch each other (sharing one or more $(d-1)$-dimensional faces) but do not overlap, so that the union of their cell sets is a valid (connected) polycube, see Figure 1 for an example in the plane. This definition generalizes for other lattice animals in a straightforward way.

*Work on this paper by the first author has been supported in part by FWF Grant P28466. Work on this paper by the second author has been supported in part by ISF Grant 575/15 and BSF Grant 2017684. Work on this paper by the third author has been supported in part by NSF Grant 1815073. A preliminary version of this paper appeared in Ref. [2].

[†]Inst. für Mathematik, Alpen-Adria-Universität Klagenfurt, Universitätsstraße 65–67, 9020 Klagenfurt am Wörthersee, Austria. E-mail: `andrei.asinowski@aau.at`

[‡]Dept. of Computer Science, The Technion—Israel Inst. of Technology, Haifa 3200003, Israel. E-mail: `barequet@cs.technion.ac.il`

[§]Dept. of Computer Science, The Technion—Israel Inst. of Technology, Haifa 3200003, Israel. E-mail: `gilbe@cs.technion.ac.il`

[¶]Dept. of Computer Science, Univ. of California, Irvine, CA 92697. E-mail: `mosegued@uci.edu`

[‖]Institut für Informatik, Freie Universität Berlin, Takustraße 9, 14195 Berlin, Germany. E-mail: `rote@inf.fu-berlin.de`

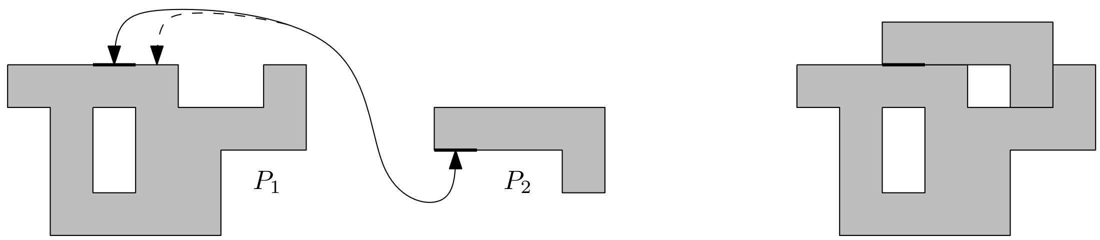


Figure 1: Aligning the edges connected by the arrow-curve creates a composition of the two polyominoes P_1 and P_2, as shown on the right. The alignment along the dotted curve does not create a valid composition, because it would lead to an overlap between P_1 and P_2.

The number of compositions plays an important role in proving bounds on the growth constant of lattice animals. For example, it was used for obtaining an upper bound on the growth constant of polyiamonds (edge-connected sets of cells on the regular planar triangular lattice) [3].[1]

In this paper we address the following.

> Question 1: Given two polycubes **of total size $2n$**, how many different compositions do they have?

We can also ask a restricted version:

> Question 2: Given two polycubes, **each of size n**, how many different compositions do they have?

Notice that all the polycubes, as well as their compositions, are considered up to translations. That is, polycubes that can be obtained from each other by a parallel translation, are considered as the same combinatorial object.

Since the situation in Question 2 is a special case of that in Question 1, some bounds for one of the questions carry over to the other question. Namely, any lower (resp., upper) bound on the minimum (resp., maximum) number of compositions in Question 1 also carries over to Question 2, and any upper (resp., lower) bound on the minimum (resp., maximum) number of compositions in Question 2 also carries over to Question 1. In fact, all our bounds apply to both versions of the question. In addition, any specific example provides both an upper bound on the minimum and a lower bound on the maximum of the respective number of compositions. We summarize our results in Table 1.

We also provide an efficient algorithm for computing the number of composition of two given polyominoes (or polycubes) (Theorem 13 in Section 5).

[1] A linear upper bound on the maximum possible number of compositions of polyominoes has been incorrectly claimed [1, Theorem 2.5], leading to an erroneous improvement of an upper bound on the growth constant of polyominoes [1, Theorem 2.6]. A correct bound on the number of compositions is given below in Theorem 4.

Table 1: The number of compositions of two polycubes of total size $2n$.

Number of Compositions	Two Dimensions		$d \geq 3$ Dimensions	
	Lower Bound	Upper Bound	Lower Bound	Upper Bound
Minimum	$\Theta(n^{1/2})$		$2n^{1-1/d}$	$O(2^d dn^{1-1/d})$
Maximum	$n^2/2^{O(\log^{1/2} n)}$	$O(n^2)$	$\Theta(dn^2)$	

2 Two Dimensions

2.1 Minimum Number of Compositions

Theorem 1. *(i) Any two polyominoes of sizes n_1 and n_2 have $\Omega((n_1+n_2)^{1/2})$ compositions.*
(ii) For every two numbers $n_1 \geq 1, n_2 \geq 1$, there is a pair of polyominoes of sizes n_1 and n_2 with $\Theta((n_1+n_2)^{1/2})$ compositions.

Proof. Let $n = n_1 + n_2$, and consider a pair of polyominoes P_1, P_2 of sizes n_1 and n_2. Assume without loss of generality that $n_1 \geq n_2$, that is, $n_1 \geq n/2$. Assume, also without loss of generality, that the width (x-span) of P_1 is greater than (or equal to) the height (y-span) of P_1. Hence, the width of P_1 is at least $n_1^{1/2}$. Then, P_2 may touch P_1 from below or above in different ways at least twice this width: Simply put P_2 below (or above) P_1 so that the left column of P_2 is aligned with the ith column of P_1 (for $1 \leq i \leq n_1^{1/2}$) and translate P_2 upward (or downward) until it touches P_1. Hence, we have a least $2n_1^{1/2} \geq (2n)^{1/2}$ compositions.

To see that this lower bound is tight, we take polyominoes that fit in a square with side lengths $k_1 = \lceil n_1^{1/2} \rceil$ and $k_2 = \lceil n_2^{1/2} \rceil$. We form P_1 and P_2 by filling the respective squares row-wise until they have the desired size. Polyominoes P_1 and P_2 can be composed in at most $4(k_1+k_2-1) \leq 4(n_1^{1/2} + n_2^{1/2} + 1) \leq 4\sqrt{2}(n_1+n_2)^{1/2} + 4$ ways. ☐

The following is a direct corollary of Theorem 1.

Corollary 2. *Any two polyominoes of total size $2n$ have $\Omega(n^{1/2})$ compositions. This lower bound is attainable.* ☐

2.2 Maximum Number of Compositions

In this section, we find bounds on the maximum number of compositions of two polyominoes of size n. First, we show a (quite trivial) upper bound of $O(n^2)$. Next, we show that it is "almost tight" by constructing an example that yields a lower bound of $\Omega(n^{2-\varepsilon})$, for any $\varepsilon > 0$.

2.2.1 Upper bound

Observation 3. *Any two polyominoes of sizes n_1 and n_2 have $O(n_1 n_2)$ compositions.*

Proof. Let n_1, n_2 denote the sizes of polyominoes P_1 and P_2, respectively. Then, every cell of P_1 can touch every cell of P_2 in at most four ways, yielding $4n_1n_2$ as a trivial upper bound on the number of compositions. For $n = n_1 + n_2$, this directly gives the bound of $O(n^2)$. ☐

2.2.2 Lower bound

It was claimed [1] that the number of compositions of two polyominoes of total size n is bounded from above by $2n$, which would be a substantial improvement of the bound $O(n^2)$ from Observation 3. Unfortunately, its proof contained an erroneous argument, and here we construct an example showing that in fact "almost" n^2 compositions are possible.

Theorem 4. *For every $n \geq 1$, there are two polyominoes, each of size at most n, that have at least*

$$\frac{n^2}{2^{8\cdot\sqrt{\log_2 n}}} \tag{1}$$

compositions.

Remarks. From now on, "log" will always denote the binary logarithm. The denominator $2^{8\cdot\sqrt{\log n}}$ grows asymptotically more slowly than x^ε for any $\varepsilon > 0$. Hence, the maximum number of compositions is $\Omega(n^{2-\varepsilon})$ for any $\varepsilon > 0$. On the other hand, if $n \leq 2^{64}$, then $8 \geq \sqrt{\log n}$, and the denominator of the bound (1) can be estimated as

$$2^{8\cdot\sqrt{\log n}} \geq 2^{\sqrt{\log n}\cdot\sqrt{\log n}} = n.$$

Hence, the claimed bound (1) is not bigger than n, which is weaker (smaller) than the number $4n$ of compositions of two $1 \times n$ "sticks." Thus, the bound in the general form (1) starts to beat the trivial bound only for *very large* values of n. The reason for this is that our analysis concentrates on getting bounds that are both *explicit* and *asymptotically strong*, at the expense of small n.

After we describe and analyze our construction, we discuss weaker bounds that can be derived from it and that exhibit superlinear growth already for moderate sizes.

Proof. We will recursively construct a series of polyominoes $D_0, D_1, D_2, \ldots$, which we call *dense toothbrushes*, and a series of polyominoes $S_0, S_1, S_2, \ldots$, which we call *sparse toothbrushes*; see Figure 2. We refer to D_k and S_k as *toothbrushes of order k*. In addition to k, these polyominoes are also parameterized by a *degree parameter*, $r \geq 2$, that indicates how many copies of toothbrushes of order $k-1$ are used to construct a toothbrush of order k. We use $r = 3$ in Figure 2. The basic building elements of toothbrushes are *sticks*—rectangles of height 1 or width 1—with one extreme cell identified as *root* and another as *apex*, so that each stick is considered to be oriented from its root to its apex. Toothbrushes D_k and S_k consist of i-sticks—sticks at *levels* $i = 0, 1, 2, \ldots, k$—where ($< k$)-sticks come recursively from toothbrushes of order $< k$, and they are attached to a "new" k-stick. The sticks cycle directions while opposing each other and have increasing lengths as shown in Table 2. (Level -1 does not exist, but it is convenient to define $\ell_{-1} = 1$.)

The toothbrushes are constructed as follows. The 0-order toothbrushes D_0 and S_0 are simply 0-sticks, *i.e.*, horizontal 1×2 dominoes, the root being the left cell for D_0, and the right cell for S_0. For $k \geq 1$, the toothbrush D_k (resp., S_k) consists of a *handle*—a k-stick of length ℓ_k, oriented as specified in Table 2—to which r copies of D_{k-1} (resp., of S_{k-1}) are attached, so that their roots coincide with the cells of the handle at distance $\alpha \cdot o_k^D$ (resp., $\alpha \cdot o_k^S$), $\alpha = 0, 1, \ldots, r-1$ cells away of its apex. The factors o_k^D, o_k^S are listed in Table 2 as the *offsets* between successive copies of D_{k-1} (resp., of S_{k-1}) along the handle of D_k (resp., of S_k). *As an exception to this rule*, the smallest dense toothbrush D_1 is constructed by attaching the copies of D_0 at distances $1, 3, 5, \ldots, 2r-1$ from the apex, instead of the distances $0, 2, 4, \ldots, 2r-2$ that would conform to the general pattern.

Table 2: Orientations and sizes of i-sticks for the recursive construction; the offsets between successive copies of D_{i-1} or S_{i-1} along the i-sticks.

Level i	Orientation of i-sticks		Stick length ℓ_i	Offset o_i^D in D_i	Offset o_i^S in S_i
	in D_i	in S_i			
(-1)			1		
0	$\rightarrow$	$\leftarrow$	2		
1	$\uparrow$	$\downarrow$	$2r^2$	2	$2r$
2	$\leftarrow$	$\rightarrow$	$4r^2$	4	$4r$
3	$\downarrow$	$\uparrow$	$4r^4$	$4r^2$	$4r^3$
4	$\rightarrow$	$\leftarrow$	$8r^4$	$8r^2$	$8r^3$
5	$\uparrow$	$\downarrow$	$8r^6$	$8r^4$	$8r^5$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
0 mod 4	$\rightarrow$	$\leftarrow$	$2^{(i+2)/2}r^i$	$2^{(i+2)/2}r^{i-2}$	$2^{(i+2)/2}r^{i-1}$
2 mod 4	$\leftarrow$	$\rightarrow$			
1 mod 4	$\uparrow$	$\downarrow$	$2^{(i+1)/2}r^{i+1}$	$2^{(i+1)/2}r^{i-1}$	$2^{(i+1)/2}r^i$
3 mod 4	$\downarrow$	$\uparrow$			

Figure 2 illustrates the construction. Dense toothbrushes are green, and sparse toothbrushes red. For dense and sparse toothbrushes of order 0 and 1, the roots are marked by blue dots. Arrows indicate the positions where the toothbrushes are attached to the handle of the next order.

As a result of these rules, sub-brushes always fan off to the *right* of the handle when viewed from the root towards the apex. As k increases, the orientation of the brushes cycles counterclockwise in the order left-down-right-up.

Thus, the difference between dense and sparse toothbrushes is that the copies of $(k-1)$-order toothbrushes are denser in D_k and sparser in S_k, and that D_0 is oriented to the right and S_0 to the left, and then similarly for higher levels: the sticks of the same level have opposite orientations in D_k and S_k.

For later reference, we record the relations between lengths and offsets from Table 2:

$$o_i^D = 2\ell_{i-2}, \qquad o_i^S = r \cdot o_i^D, \qquad \ell_i = r \cdot o_i^S = 2r^2\ell_{i-2}. \tag{2}$$

As a consequence, one can observe that when we increase the level i by two steps, all dimensions increase by a factor of $2r^2$.

The 0-sticks consist of two squares, but since one of these squares overlaps a vertical 1-stick, they appear as single-square protrusions, or *notches*. These notches will play a crucial role in counting the compositions. Each of the toothbrushes D_k and S_k has r^k notches. We represent each notch N of D_k by a sequence $A = (\alpha_1, \alpha_2, \dots, \alpha_k)$, where α_i indicates that the copy of D_{i-1} that contains N is attached to the level-i handle at distance $\alpha_i o_i^D$ from its apex (or for $i = 1$, at distance $1 + \alpha_i o_i^D = 1 + 2\alpha_1$). The "digits" α_i of this representation (for $1 \le i \le k$) are in the range $0 \le \alpha_i \le r-1$. We also use a similar encoding $B = (\beta_1, \beta_2, \dots, \beta_k)$ for notches of S_k. In Figure 2, two notches are marked by crosses: the notch $(2, 0, 2, \dots)$ of (green) D_k and the notch $(1, 2, 2, \dots)$ of (red) S_k.

Lemma 5. *The size of D_k and S_k is bounded from above by $2^{(k+2)/2}r^{k+1}\left(1 + \frac{2}{r}\right)$ for even k, and*

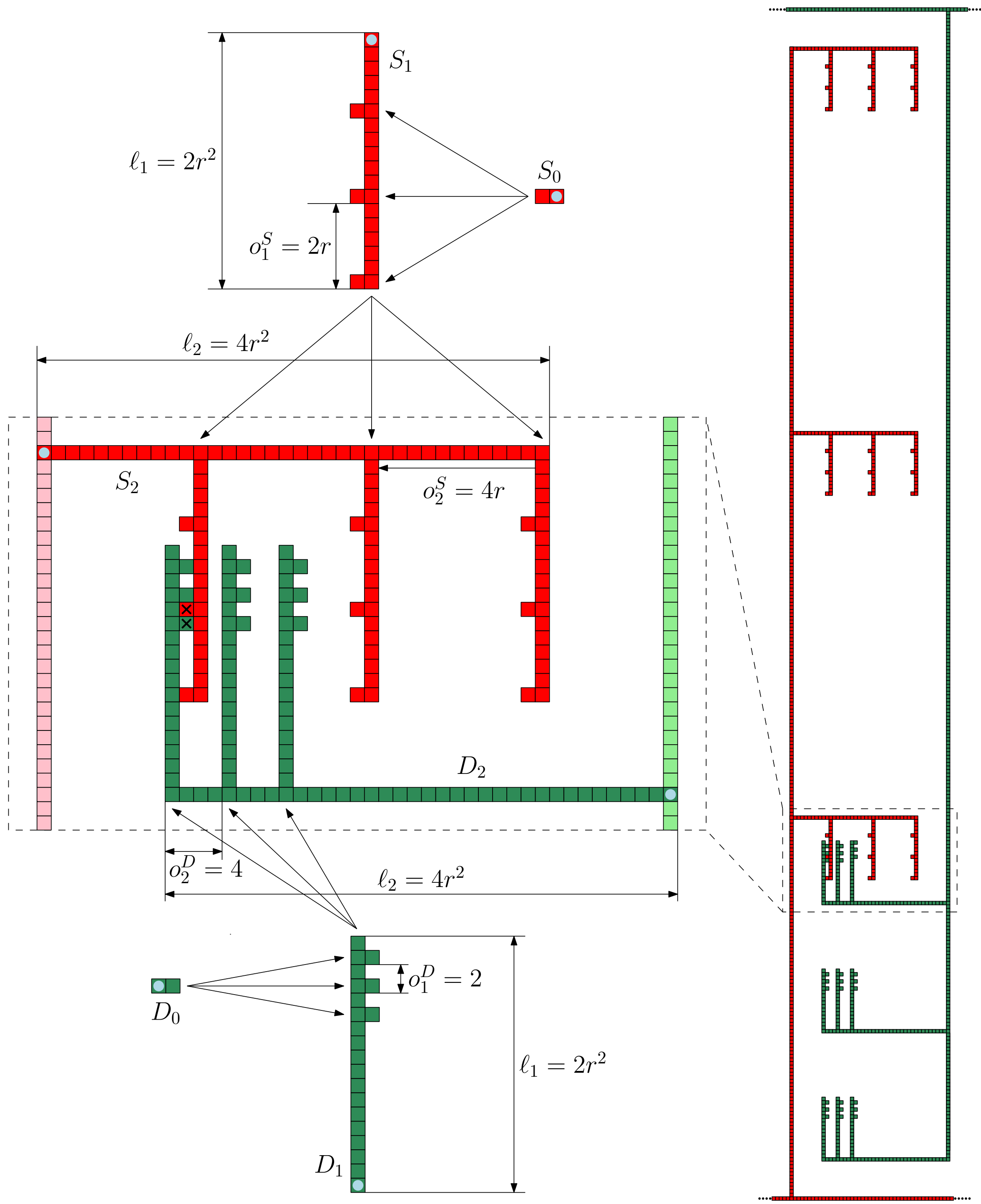


Figure 2: The construction for $r = 3$. The roots of $D_0, S_0, D_1, S_1, D_2, S_2$ are marked with blue dots.

by $2^{(k+3)/2}r^{k+1}\big(1+\frac{1}{r}\big)$ *for odd* k*. A common upper bound for both cases is*

$$3\big(\sqrt{2}\cdot r\big)^{k+1}. \tag{3}$$

Proof. To get an upper bound, we simply add the sizes of all sticks, ignoring the overlaps. Let us begin with k being even. The handle of D_k or S_k is horizontal and has size $2^{(k+2)/2}r^k$. There are r copies of D_{k-1} or S_{k-1}, and their r vertical handles have total size $r\times 2^{k/2}r^k$. Together, the sticks at the top two levels have size

$$2^{k/2+1}r^k + 2^{k/2}r^{k+1} = 2^{k/2}r^{k+1}\big(1+\tfrac{2}{r}\big). \tag{4}$$

When going down two levels, the stick length decreases by a factor of $2r^2$, but the number of sticks increases by a factor of r^2. Thus, the total size of the sticks decreases by a factor of 2. Counting separately the sticks at even and at odd levels, we therefore get an upper bound on the total size of all sticks if we multiply (4) by $1+\frac{1}{2}+\frac{1}{4}+\cdots=2$. This proves the first statement.

For odd k, we obtain in a similar way

$$\Big(2^{(k+1)/2}r^{k+1} + r\times 2^{(k+1)/2}r^{k-1}\Big)\times\big(1+\tfrac{1}{2}+\tfrac{1}{4}+\cdots\big) = 2^{(k+3)/2}r^{k+1}\big(1+\tfrac{1}{r}\big).$$

The factor 3 in (3) is large enough to cover the extra term $\sqrt{2}\times\big(1+\frac{2}{r}\big)\le\sqrt{2}\times 2$ for the even case and $2\times\big(1+\frac{1}{r}\big)\le 2\times\frac{3}{2}$ for the odd case. □

Lemma 6. *There are at least* r^{2k} *compositions of* D_k *and* S_k*.*

Proof. For each notch N_D of D_k and for each notch N_S of S_k, we can translate D_k and S_k so that the upper edge of N_D coincides with the lower edge of N_S. In the inset of Figure 2, the two involved notches are marked by crosses.

We claim that (1) Such r^{2k} compositions are distinct; and (2) Each of them is valid in the sense that D_k and S_k positioned in this way are disjoint. (We ignore many other compositions, but asymptotically, this gives the dominant term of the total number of compositions.)

(1) We first argue that all these compositions are distinct. Let N_D be a notch of D_k represented by a sequence $A=(\alpha_1,\alpha_2,\dots,\alpha_k)$, as explained above. Let us position D_k so that the notch encoded by $(0,0,\dots,0)$ has coordinates $\binom{0}{0}$. Then, the coordinates of the notch N_D are

$$\begin{pmatrix}0\\0\end{pmatrix}+\alpha_1\begin{pmatrix}0\\-o_1^D\end{pmatrix}+\alpha_2\begin{pmatrix}o_2^D\\0\end{pmatrix}+\alpha_3\begin{pmatrix}0\\o_3^D\end{pmatrix}+\alpha_4\begin{pmatrix}-o_4^D\\0\end{pmatrix}+\cdots=$$
$$\begin{pmatrix}4\cdot\alpha_2-8r^2\cdot\alpha_4+16r^4\cdot\alpha_6-32r^6\cdot\alpha_8+\dots\\-2\cdot\alpha_1+4r^2\cdot\alpha_3-8r^4\cdot\alpha_5+16r^6\cdot\alpha_7-\dots\end{pmatrix}. \tag{5}$$

If we similarly encode the notch N_S of S_k by $B=(\beta_1,\beta_2,\dots,\beta_k)$ and position S_k so that the notch encoded by $(0,0,\dots,0)$ has coordinates $\binom{0}{0}$, then the coordinates of the notch N_S are

$$\begin{pmatrix}0\\0\end{pmatrix}+\beta_1\begin{pmatrix}0\\o_1^S\end{pmatrix}+\beta_2\begin{pmatrix}-o_2^S\\0\end{pmatrix}+\beta_3\begin{pmatrix}0\\-o_3^S\end{pmatrix}+\beta_4\begin{pmatrix}o_4^S\\0\end{pmatrix}+\cdots=$$
$$\begin{pmatrix}-4r\cdot\beta_2+8r^3\cdot\beta_4-16r^5\cdot\beta_6+32r^7\cdot\beta_8-\dots\\2r\cdot\beta_1-4r^3\cdot\beta_3+8r^5\cdot\beta_5-16r^7\cdot\beta_7+\dots\end{pmatrix}. \tag{6}$$

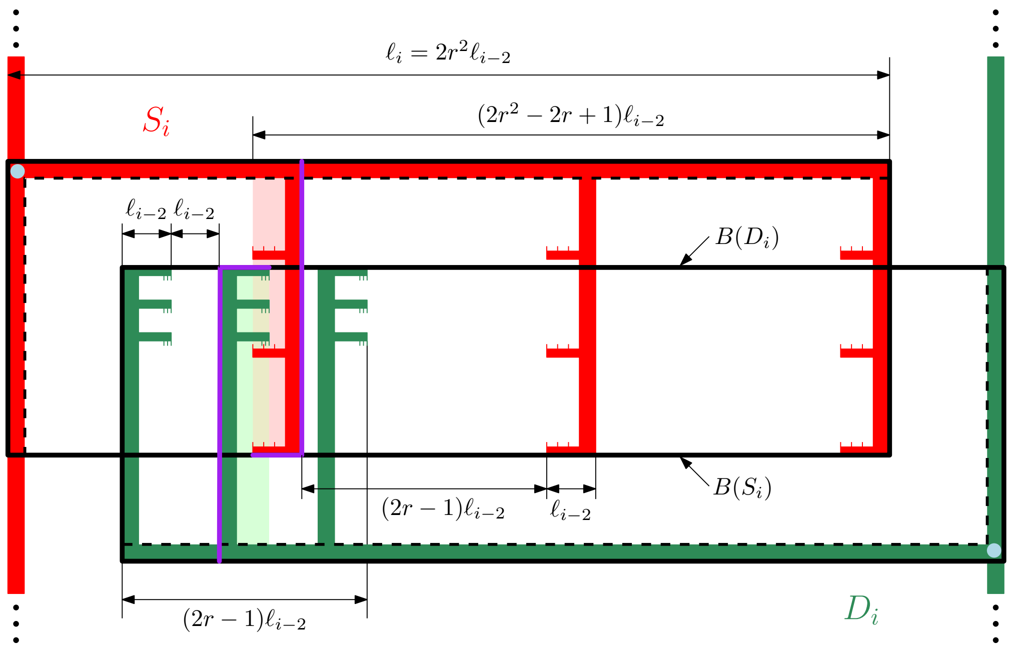


Figure 3: Schematic illustration of bounding boxes and rims for toothbrushes D_i and S_i.

The translation of S_k that brings N_S to the cell directly above N_D is found by taking the difference between Equations (5) and (6), and adding $\binom{0}{1}$:

$$\begin{pmatrix} 4\alpha_2 + 4r\beta_2 - 8r^2\alpha_4 - 8r^3\beta_4 + \dots \\ 1 - 2\alpha_1 - 2r\beta_1 + 4r^2\alpha_3 + 4r^3\beta_3 - \dots \end{pmatrix}.$$

Since both the successive multipliers $4, 4r, 8r^2, 8r^3, \dots$ for the x-coordinate and the successive multipliers $2, 2r, 4r^2, 4r^3, \dots$ for the y-coordinate differ at least by a factor of r, and the coefficients α_i and β_i are between 0 and $r-1$, we conclude that distinct $2r$-tuples $(a_1, a_2, \dots, a_r, b_1, b_2, \dots, b_r)$ lead to distinct translations.

(2) It remains to prove that the 2^{2k} compositions described above are valid. That is, to show that if we translate D_k and S_k so that some notch N_D of D_k is just below some notch N_S of S_k, then the union of D_k and S_k is disjoint. This will be accomplished by the following Claims 7 and 8.

For each polyomino P, let its *bounding box* $\mathsf{B}(P)$ be the smallest (filled) grid rectangle that contains it. It is easy to see that the bounding boxes of D_i and of S_i have size $\ell_i \times \ell_{i-1}$ or $\ell_{i-1} \times \ell_i$ (using the convention $\ell_{-1} = 1$). We define the *rim* of a toothbrush as the union of the sides—one horizontal and one vertical—of its bounding box that contain the root of its handle. In fact, one of the sides of the rim of an i-order toothbrush is its handle, and the other side is contained in the handle of the $(i+1)$-order toothbrush to which it belongs. In Figure 3, bounding boxes of two toothbrushes are shown by bold frames, and the bending point of the respective rims are marked by a blue dot. Bounding boxes of some toothbrushes of smaller order are shown by light green or pink background. One should keep in mind that this figure is schematic and sticks of different levels are not to scale.

Claim 7. *Consider a composition of D_k and S_k as described above. Let $1 \le i \le k$, and suppose that the composition is established via the notches N_D of D_k and N_S of S_k.*[2] *Suppose further that N_D*

[2] Recall that this means that N_D is just below N_S.

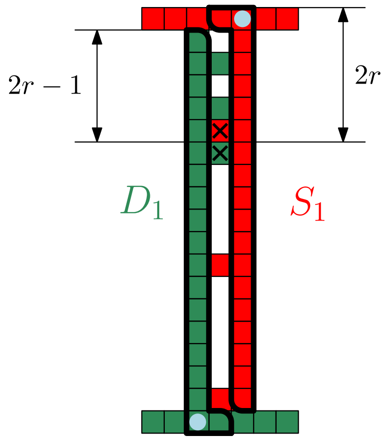


Figure 4: Proof of Claim 7, case $i = 1$.

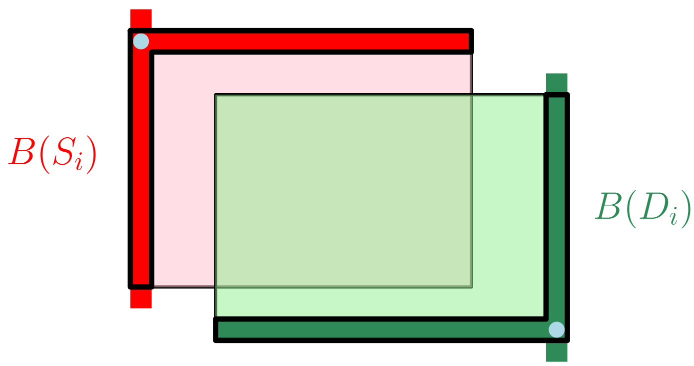


Figure 5: Illustration of Claim 7: Bounding boxes overlap, but the rims never overlap. Since the situation is symmetric, it is sufficient to prove the claim for one of the rims.

lies in some copy of D_i and the notch N_S lies in some copy of S_i. Then the bounding boxes $\mathsf{B}(D_i)$ and $\mathsf{B}(S_i)$ overlap, but neither bounding box overlaps the rim of the other toothbrush. (Refer to Figure 5 for a schematic depiction of the statement.)

Proof. We prove the claim by induction. For $i = 1$, it is easily checked by inspection; refer to Figure 4. The notches do not overlap since all the notches of D_1 fit into gaps between notches in S_1. It remains to show that $\mathsf{B}(D_1)$ cannot reach the uppermost row of $\mathsf{B}(S_1)$. Indeed, if N_D is the lowest notch of D_1, the vertical distance from its upper edge to the top of $\mathsf{B}(D_1)$ is $r \cdot o_1^D - 1 = 2r - 1$. If N_S is the highest notch of S_1, the vertical distance from its lower edge to the top of $\mathsf{B}(S_1)$ is $o_1^S = 2r$. Thus, if the upper edge of N_D coincides with the lower edge of N_S, the top of $\mathsf{B}(D_1)$ is still strictly below the top of $\mathsf{B}(S_1)$.

Now let $i \geq 2$. Assume without loss of generality that the rim of D_i occupies the lower and the right side of $\mathsf{B}(D_i)$, and the rim of S_i occupies the upper and the left side of $\mathsf{B}(S_i)$, as shown by bold frames in Figure 3. Let D_{i-1} and S_{i-1} be specific copies of the lower-order toothbrushes that contain the notches N_D and N_S. Their bounding boxes are shown in the figure with a shaded background. Since $\mathsf{B}(D_{i-1})$ and $\mathsf{B}(S_{i-1})$ overlap by induction, we immediately get the overlap of $\mathsf{B}(D_i)$ and $\mathsf{B}(S_i)$.

To prove that the rim of S_i does not overlap $\mathsf{B}(D_i)$, we need to show that $\mathsf{B}(D_i)$ can reach neither the highest row nor the leftmost column of $\mathsf{B}(S_i)$. The former claim is easy: The rim of S_{i-1} contains the highest row of $\mathsf{B}(S_{i-1})$, and by induction, $\mathsf{B}(D_{i-1})$ does not overlap with this row. The box $\mathsf{B}(D_i)$ uses the same rows as $\mathsf{B}(D_{i-1})$, and similarly for $\mathsf{B}(S_i)$ and $\mathsf{B}(S_{i-1})$. Therefore, $\mathsf{B}(D_i)$ cannot reach the highest row of $\mathsf{B}(S_i)$.

To show that $\mathsf{B}(D_i)$ cannot reach the leftmost column of $\mathsf{B}(S_i)$, we use the relations (2) in the calculation. The horizontal extension of each $(i-1)$-order sub-brush D_{i-1} or S_{i-1} is ℓ_{i-2}. The $(i-1)$-order sub-brushes of D_i span in total a horizontal range of width $(r-1)o_i^D+\ell_{i-2} = (2(r-1)+1)\ell_{i-2}$, starting to the right from the left side of $\mathsf{B}(D_i)$. The $(i-1)$-order sub-brushes of S_i span in total a horizontal range of width $(r-1)o_i^S + \ell_{i-2} = (2r(r-1)+1)\ell_{i-2}$, starting to the left from the right side of $\mathsf{B}(S_i)$. The sum of these two distances is just equal to the horizontal extension of D_i and S_i: $\ell_i = 2r^2\ell_{i-2}$. It follows that $\mathsf{B}(D_i)$ cannot reach the leftmost column of $\mathsf{B}(S_i)$ if the bounding boxes of some $(i-1)$-order sub-brushes overlap, which holds by induction for the specified copies of D_{i-1} and S_{i-1}. ☐

Claim 8. *Consider two (sub-)brushes D_i and S_i of order $i \geq 2$. If two of their sub-brushes D_{i-1} and S_{i-1} have overlapping bounding boxes, then no other pair of sub-toothbrushes D'_{i-1} and S'_{i-1} of order $i-1$ can have overlapping bounding boxes.*

Proof. We employ the same assumption as for the orientation of D_i and S_i as in the previous proof. The horizontal dimension of the bounding boxes of level $i-1$ is then ℓ_{i-2}. The offset between different copies of D_{i-1} is $o_i^D = 2\ell_{i-2}$, by (2). Hence, the distance between their bounding boxes is ℓ_{i-2}, and, therefore, no toothbrush S_{i-1} can intersect with two different copies of D_{i-1}.

We also have to argue that no two copies of S_{i-1} can be intersected by some D_{i-1}. The offset between successive copies of S_{i-1} is $o_i^S = 2r\ell_{i-2}$, and, hence, the gap between their bounding boxes is $(2r-1)\ell_{i-2}$. On the other hand, all copies of D_{i-1} together fit in a box of horizontal extension $(2r-1)\ell_{i-2}$. Hence, no toothbrush D_{i-1} can intersect with two different copies of S_{i-1}. ☐

With Claims 7 and 8, we can now conclude that D_k and S_k are disjoint: It follows from Claim 7 that the handle of D_k is disjoint from S_k (even from its bounding box), and vice versa. All cells that are not in the handle are in the sub-brushes D_{k-1} and S_{k-1}. There is exactly one pair D_{k-1}, S_{k-1} that contains N_D and N_S, respectively, and by Claim 7, the respective bounding boxes overlap. By Claim 8, this means that all other pairs S'_{k-1}, L'_{k-1} are disjoint. It suffices, therefore, to prove the claim for sub-brushes D_{k-1} and S_{k-1} that contain N_D and N_S.

However, the proof above applies for sub-brushes of any order. In this way, we proceed by induction to toothbrushes of lower order until we reach the order-0 pair D_0, S_0 containing the notches N_D and N_S, for which disjointness is obvious. This concludes the proof of Lemma 6. ☐

In order to finish the proof of Theorem 4, we apply the construction with the parameters $k := \lfloor\sqrt{\log n}\rfloor - 1$ and $r := 2^k$. We assume[3] that $n \geq 16$, hence $k \geq 1$ and $r \geq 2$.

We use Lemma 5 to show that the size of the polyominoes is at most n. The logarithm of the

[3] Recall the discussion after the statement of the theorem. It is shown there that for $n \leq 2^{64}$, our construction does not beat the trivial bound.

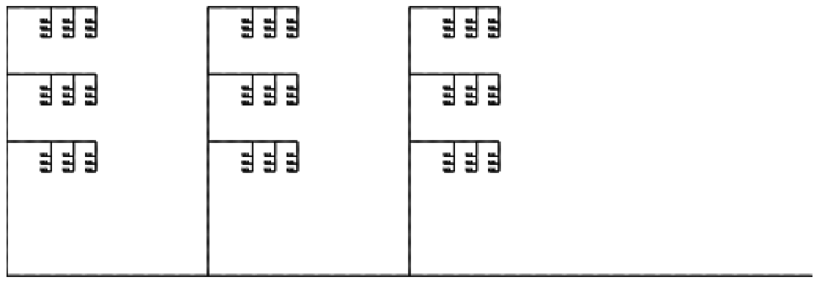


Figure 6: A rendering of a variation of our sparse toothbrush S_5 as an L-system.

bound (3) is

$$\begin{aligned}
\log\big((\sqrt{2}\cdot r)^{k+1}3\big) &= \log\big((\sqrt{2}\cdot 2^k)^{k+1}\big) + \log 3 \\
&= (k+1/2)(k+1) + \log 3 \\
&\le (\sqrt{\log n} - 1/2)\sqrt{\log n} + \log 3 \\
&= \log n - \sqrt{\log n}/2 + \log 3 \\
&\le \log n,
\end{aligned}$$

where the last relation holds for $n \ge 1059$.

Now we apply Lemma 6 in order to estimate the number of compositions from below. Again, we compute the logarithm of the desired quantity:

$$\log(r^{2k}) = \log\big((2^k)^{2k}\big) = 2k^2 \ge 2(\sqrt{\log n} - 2)^2 = 2\log n - 8\sqrt{\log n} + 8 \ge 2\log n - 8\sqrt{\log n}.$$

From this, we directly get the bound (1). □

There are a few obvious local improvements of our construction. For example, the necessary spacing between the level-1 vertical sticks in D_2 is only 3 instead of the 4 that we use. Removing all notches allows to reduce the spacing even further, without reducing the number of compositions that we count. Alternatively, we could replace the notches by sticks of length r and adjust all horizontal dimensions accordingly. This would increase the number of compositions by the factor $r-1$, while increasing the sizes only by a constant factor. By contrast, our proof strives to make the description of the construction as easy as possible and to keep simple expressions for the dimensions in terms of powers of 2 and r.

By choosing a small constant order k, we already obtain superlinear bounds from Lemmata 5 and 6. For example, $k = 3$ leads to toothbrushes of size $n = O(r^4)$ with at least r^6 compositions, *i.e.*, $\Omega(n^{3/2})$ compositions. Setting $k = 4$ leads to toothbrushes of size $n = O(r^5)$ with at least r^8 compositions, *i.e.*, $\Omega(n^{8/5})$ compositions, etc. For any fixed k, we get $\Omega(n^{2-2/(k+1)})$ compositions.

Remark: As $k \to \infty$, the toothbrushes D_k and S_k, properly scaled and rotated, converge to tree-like structures whose substructures are "similar" to the whole structure: thus, it bears some similarity to fractals. The limits are different for D_k and S_k, and, in addition, we have to distinguish between even and odd values of k. When going down two orders, all lengths are uniformly scaled by $1/(2r^2)$, and, hence, we find self-similar substructures. However, since the number of substructures is only r^2, the total length is finite, and the fractal dimension is 1. Hence, we don't have a fractal in the strict sense. We mention that our toothbrushes, like many fractals, can be modeled by L-systems,[4] for example, as follows:

```
Constants: X
Axiom:     --X
Rule1:     X=[-FFXFXFX]
Rule2:     F=FFF
```

An L-system renderer (`http://www.kevs3d.co.uk/dev/lsystems/`) produces, using the specification above, the fractal shown in Figure 6. In this L-system, a string of symbols is converted to an image by interpreting the symbols as turtle graphics commands: The letter `F` makes a step forward, and the symbol '`-`' makes a right turn by 90°. The symbol '`[`' saves the current position and orientation on a stack, and '`]`' returns to the previously saved state. The letter `X` is ignored for the drawing. In one iteration, all occurrences of `X` and `F` in the current string are simultaneously substituted according to the two rules. Figure 6 is produced from the starting string (axiom) "`--X`" after 6 iterations.

We note that the fractal dimension [7] is not the relevant parameter for our problem since it measures the length of a fractal curve (the boundary of the polyomino, in our setting) in terms of the diameter. However, for our application, we also want the *size* (the area enclosed by the boundary) to be small.

3 Higher Dimensions

3.1 Minimum Number of Compositions

3.1.1 Lower bound

Theorem 9. *Any two d-dimensional polycubes of total size $2n$ have at least $2n^{1-1/d}$ compositions.*

Proof. The proof is similar to that of Theorem 1. Consider two polycubes P_1, P_2 of total size $2n$. Assume, without loss of generality, that P_1 is the larger of the two polycubes, that is, the size (d-dimensional volume) of P_1 is at least n. Let V_i (for $1 \le i \le d$) be the $(d-1)$-dimensional volume of the projection of P_1 orthogonal to the x_i axis. An isoperimetric-like inequality of Loomis and Whitney [6] ensures that $\prod_{i=1}^{d} V_i \ge n^{d-1}$. Let $V_k \ge n^{1-1/d}$ be largest among the numbers $V_1, \ldots, V_d$. Then, there are at least $2V_k \ge 2n^{1-1/d}$ different ways for how P_2 may touch P_1. The polycube P_1 has V_k "columns" in the x_k direction. Pick one specific such "column" of P_2 and align it with each "column" of P_1, putting it either "below" or "above" P_1 along direction x_k, and find the unique translation along x_k by which they touch for the first time while being translated one towards the other. □

3.1.2 Upper bound

Theorem 10. *There exist pairs of d-dimensional polycubes, of total size $2n$, that have $O(2^d d n^{1-1/d})$ compositions.*

[4] `https://en.wikipedia.org/wiki/L-system`

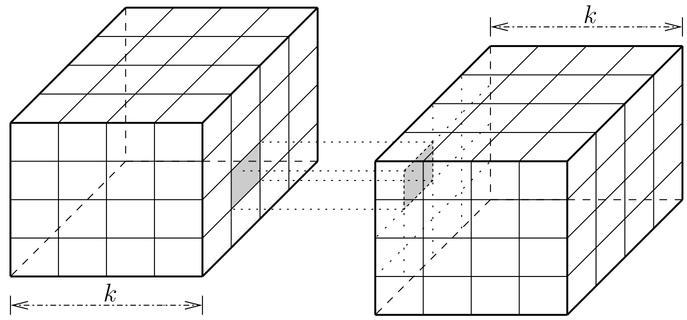
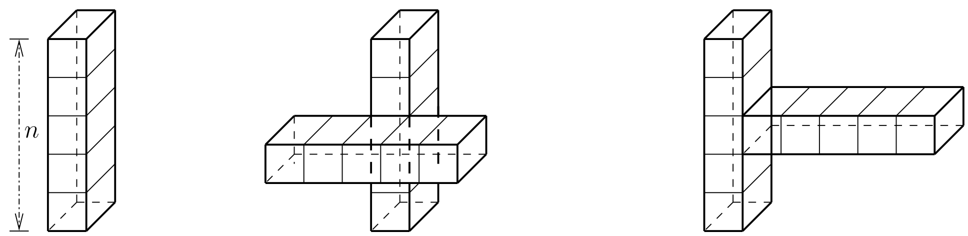


Figure 7: A composition of two hypercubes.



(a) A stick (b) A side-facet composition (c) An extreme-facet composition

Figure 8: Compositions of two "sticks."

Proof. Figure 7 shows a composition of two copies of a d-dimensional hypercube P of size $k \times k \times \ldots \times k$ ($d = 3$ in the figure). The cube is made of n cells, hence, its sidelength is $k = n^{1/d}$. Two copies of P can slide towards each other in $2d$ directions (two directions in each dimension) until they touch. Obviously, there are no other compositions since no hypercube can penetrate into the bounding box of the other one. Once we decide which facets of the hypercube touch each other, this can be done in $(2k{-}1)^{d-1}$ ways. Indeed, in each of the $d{-}1$ dimensions orthogonal to the sliding direction, there are $2k{-}1$ possible offsets of one hypercube relative to the other. (This can be visualized easily in two and three dimensions.) Overall, the total number of compositions in this example is

$$(2d)(2k-1)^{d-1} = 2d(2n^{1/d}-1)^{d-1} = \Theta(2^d dn^{1-1/d}).$$

□

3.2 Maximum Number of Compositions in $d \geq 3$ Dimensions

Theorem 11. *Let $d \geq 3$. Any two d-dimensional polycubes of total size $2n$ have $O(dn^2)$ compositions. For $d \geq 3$, the upper bound is attainable: There are two d-dimensional polycubes of total size $2n$ with $\Omega(dn^2)$ compositions.*

Proof. Similarly to two dimensions, any two polycubes P_1, P_2 of total size $2n$ have $O(dn^2)$ compositions. Indeed, let n_1, n_2 denote the sizes of P_1 and P_2, respectively, where $n_1 + n_2 = 2n$. Then, every cell of P_1 can touch every cell of P_2 in at most $2d$ ways, yielding $2dn_1n_2 \leq 2dn^2$ as a trivial upper bound on the number of compositions.

The lower bound is attained asymptotically, for example, by two nonparallel "sticks" of size n, as shown in Figure 8(a). Each stick has two *extreme* $(d{-}1)$-dimensional facets (orthogonal to the direction along which the stick is aligned), plus $2(d-1)n$ many $(d-1)$-dimensional *side* facets. The number of compositions that involve only side facets is $2(d-2)n^2 = \Omega(dn^2)$, see Figure 8(b): Indeed, for each of the $d-2$ coordinate directions that are not parallel to one of the sticks, there are $2n^2$ different choices for letting two side facets of the sticks touch. We can ignore the small number of $4n$ compositions that involve an extreme facet, see Figure 8(c). □

Note the difference, for the maximum number of compositions, between the cases $d = 2$ and $d > 2$. If $d > 2$, the dimensions along which the sticks are aligned, restrict the compositions of the sticks, but the existence of more dimensions allows every pair of cells, one of each polycube, to have compositions through this pair only. This is not the case in two dimensions, a fact that makes the proof of Theorem 4 much more complicated.

4 Compositions and the Minkowski Sum

As a preparation for the algorithm that determines (or counts) all compositions, we discuss an elementary connection between compositions of two polyominoes and the Minkowski sum, the element-wise sum of two sets of vectors A and B:

$$A \oplus B := \{\, a + b \mid a \in A, b \in B \,\}.$$

In this connection, it is better to regard a polyomino as a discrete set A of *points*, namely the centers of the grid squares of which it is composed. The polyomino itself can then be obtained as the Minkowski sum of A with a unit square U centered at the origin: $A \oplus U$.

We call an integer vector $t \in \mathbb{Z}^d$ a *valid composition vector*, or simply a *valid composition*, if P and $Q + t$ form a valid composition, *i.e.*, they do not overlap, but share at least one common edge.

Observation 12. *Let P_1, P_2 be polyominoes and let A_1, A_2 be their sets of centerpoints.*

1. *The set of (integer) translations t for which P_1 and $P_2 + t$ overlap is the Minkowski difference*

$$F := A_1 \oplus (-A_2) := \{\, c_1 - c_2 \mid c_1 \in A_1, c_2 \in B \,\}.$$

   We call F the forbidden set.

2. *The set of valid composition vectors for P_1 and P_2 is the set of* neighbors *of F: those integer vectors that have distance 1 from a point of F but that do not themselves belong to F.*

See Figure 9 for an example.

Proof. The first statement is obvious: A vector t is of the form $t = c_1 - c_2$ for some $c_1 \in A_1$ and $c_2 \in A_2$ if and only if the cells $c_1 \in A_1$ and $c_2 + t \in A_2 + t$ coincide: $t = c_1 - c_2 \iff c_1 = c_2 + t$.

To see the second statement, let $t \notin F$ be a vector and $t' \in F$ an adjacent vector. Then, $c_1 \in A_1$ and $c_2 + t' \in A_2 + t'$ coincide. If we move $A_2 + t'$ by one unit to $A_2 + t$, the cell $c_2 + t \in A_2 + t$ is adjacent to $c_2 + t' = c_1 \in A_1$, but $A_2 + t$ becomes disjoint from A_1, and hence t is a valid composition.

On the other hand, if t is a valid composition, then $t \notin F$, but there must be two adjacent cells $c_2 + t \in A_2 + t$ and $c_1 \in A_1$. Moving A_2 by one unit brings these two cells to coincide; hence, there is a vector t' adjacent to t such that $c_2 + t' = c_1$, or in other words, $t' \in F$. □

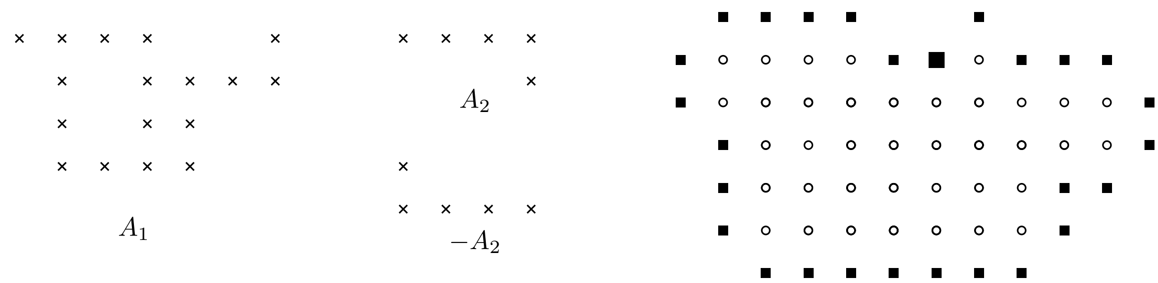


Figure 9: The sets A and B of cell centers of the polyominoes P_1 and P_2 from Figure 1, the Minkowski difference $F = A \oplus (-B)$ (circles), and the set of valid composition vectors (squares). P_1 and P_2 have 27 compositions. The composition from Figure 1 is highlighted.

5 Counting Compositions

We now describe an efficient algorithm for finding all compositions of two polyominoes or polycubes. We assume the unit-cost model of computation, in which numbers in the range $[-n, n]$ can be accessed and be subject to arithmetic operations in $O(1)$ time, and up to $O(n^2 d)$ memory cells can be accessed by their address in $O(1)$ time.

Theorem 13. *(i) Given two polyominoes, each of size at most n, their number of compositions can be computed in $O(n^2)$ time and $O(n^2)$ space.*
(ii) Given two d-dimensional polycubes, each of size at most n, their number of compositions can be computed in $O(d^2n^2)$ time and $O(dn^2)$ space.

Proof. A straightforward approach would try all $O(n^2)$ possibilities of moving a cell $y \in P_2$ next to a cell $c_1 \in P_1$, in $2d$ possible ways, and check whether the two translated polyominoes overlap. Testing for overlap can be done very naively in $O(n^2)$ time, or with little effort in $O(n)$ time, but even this leads to an overall runtime of $O(n^3)$.

However, we can do better, by using the connection to the Minkowski sum from Observation 12. Let us first deal with the situation in the plane ($d = 2$ dimensions). To compute F, we can use a bitmap data structure T, which holds the status of all possible translations in a $(2n+3) \times (2n+3)$ array, with indices in the range $-n-1 \le t \le n+1$ in each direction. Initially, all entries of T are cleared. In a double loop over the pairs of cells $c_1 \in P_1, c_2 \in P_2$, we set the entry in T corresponding to the translation $t = c_1 - c_2$. This sets the bits of F.

Obviously, both the size and preparation time of T are $O(n^2)$. Finally, by scanning each cell of T, we can determine in constant time if it lies outside F but has a neighbor belonging to F, and hence, according to Observation 12, represents a valid composition. Overall, the entire process requires $\Theta(n^2)$ time and space.

These bounds assume the worst case, in which size-n polyominoes have an extent of $\Theta(n)$ in each dimension. By contrast, typical polyominoes can be expected to be somewhat compact. However, we are not aware of any empirical evidence for this statement.

Finally, let us list the differences needed for following the same approach in d dimensions. Each cell now has d coordinates (instead of two), and so every cell or translation operation (*e.g.*, setting, comparing, checking, etc.) requires $\Theta(d)$ instead of constant time. Instead of four neighboring cells, each polycube cell now has $2d$ neighbors. The size of the input is $\Theta(dn)$. A bitmap would require space $\Theta(n^d)$, and we would like to avoid this exponential growth in d.

Instead, we will identify the cells of F by sorting. We generate the at most n^2 elements of the Minkowski difference $P_1 - P_2$, one at a time, in $O(n^2 d)$ time, and store them in a list. Then we sort this list, using radix sort. Radix sort sorts the list in d passes over the data, each time assigning the elements to buckets according to one selected digit (coordinate). Each pass takes $O(n^2)$ time (plus $O(n)$ time for the range of values of the ith coordinate). Thus, in $O(n^2 d)$ time, we get the elements of F in sorted order, and then it is easy to eliminate duplicates.

In the second step, we generate $2d$ neighbors of each element of F. These are $O(n^2 d)$ *candidates* for translations that may lead to valid compositions. We have to remove the candidates that belong to F, because they lead to collisions, and we have to eliminate duplications. Again, we rely on radix-sort, but in order to save space, we use a special representation: Each neighbor of an element x of F is represented as a triplet (x, i, s). The first component is a pointer to x. The index i lies in the range $1 \le i \le d$ and indicates which coordinate is to be incremented ($s = 1$) or decremented ($s = -1$). This representation requires only constant space per candidate neighbor, and nevertheless, it is possible to access each coordinate in constant time.

In total, we need $O(n^2 d)$ space: $O(d)$ space for each of the $O(n^2)$ elements of F, which are represented explicitly; and $O(1)$ space for each of the $O(n^2 d)$ candidates. We sort F plus the list of all candidates, using radix sort, in $O(n^2 d^2)$ time. This brings all elements with the same coordinates together, and allows us to eliminate duplicate or invalid candidates.[5] □

We mention that our algorithm actually *generates* all valid compositions within the same runtime, in the sense that some procedure can *visit* every composition once, for example in order to collect some statistics. If one insists on producing an *explicit list* of all compositions, the storage requirement might increase to $\Omega(d^2 n^2)$: By Theorem 11, there can be inputs with $\Omega(dn^2)$ compositions, each requiring size $\Theta(d)$ to write down.

6 Distribution and Average in Two Dimensions

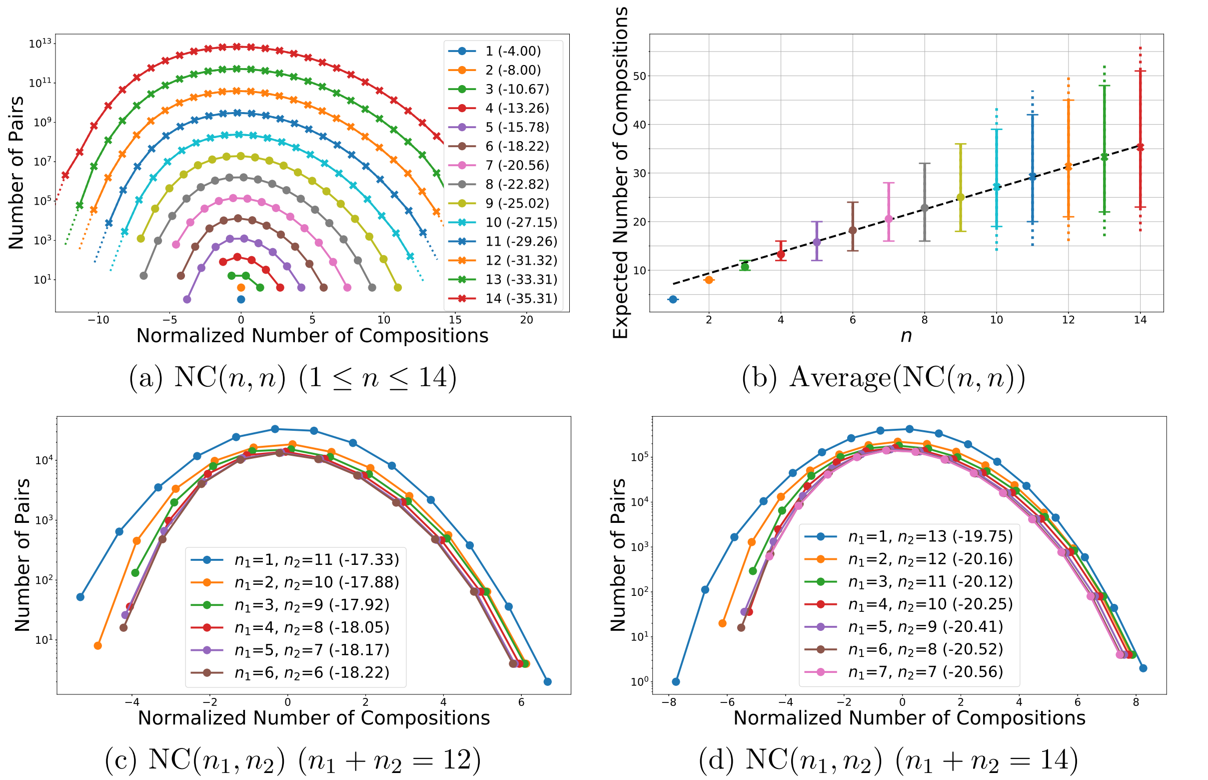
In this section, we present some empirical data concerning the interesting question of the *distribution* of $\mathrm{NC}(n_1, n_2)$, the number of compositions of all pairs of polyominoes of sizes n_1, n_2.

Figure 10(a) shows with filled circles the distributions of the number of compositions of pairs of polyominoes of the same size. For each size up to $n = 9$, we took all pairs P_1, P_2 of polyominoes of size n and counted the number of their compositions. For each number p of compositions, the graph shows the multiplicity with which p occurs, *i.e.*, the number of pairs (P_1, P_2) among the $A(n)^2$ pairs that have p compositions, on a logarithmic scale. The points for a given size n are connected by a curve. In order to make the curves for different values of n comparable, we normalized the number p by subtracting the average number of compositions for size n. Thus, the horizontal axis is actually the deviation of p from the average. (This average is shown in Figure 10(b).)

---

[5] In theory, one could combine the two phases, the generation of the elements of F, and of their neighbors, into one step without affecting the worst-case running time bound. In practice, however, eliminating duplications in F will reduce the number of elements that need to be considered in the second phase.

In the conference version of this paper [2], various algorithms with larger space complexity were discussed: Representation of F as a trie (digital search tree), in which the nodes are represented as arrays ($O(d^2 n^2)$ time and $O(dn^3)$ space) or as binary search trees ($O(d^2 n^2 \log n)$ time and $O(d^2 n^2)$ space), or a representation with hash tables ($O(d^2 n^2)$ expected time and $O(d^2 n^2)$ space).



(a) $\mathrm{NC}(n,n)$ $(1 \le n \le 14)$ (b) $\mathrm{Average}(\mathrm{NC}(n,n))$

(c) $\mathrm{NC}(n_1,n_2)$ $(n_1+n_2=12)$ (d) $\mathrm{NC}(n_1,n_2)$ $(n_1+n_2=14)$

Figure 10: Distributions of the number of compositions of pairs of polyominoes of sizes n_1, n_2. Numbers in parentheses are values by which the curves are normalized (shifted horizontally to the left).

For polyominoes of size $10 \le n \le 14$, we sampled uniformly $s = 5 \cdot 10^7$ out of all $A(n)^2$ pairs because considering all pairs of polyominoes would be too time consuming. The obtained results were multiplied by $A(n)^2/s$ in order to get an estimate for the true multiplicities. These samples represent only a small fraction of all pairs: roughly $1.7 \cdot 10^{-4}$ for $n = 10$ and $1.1 \cdot 10^{-8}$ for $n = 14$. Nevertheless, the estimates (shown with crosses in Figure 10(a)) appear visually consistent with the exact results, except that the sampling missed numbers of compositions with too few realizing pairs of polyominoes. The data were fitted to various discrete distributions, using the statistics module of the Python package `scipy`. The best fit was found with the negative-binomial distribution.

Figure 10(b) plots the average number of compositions of a pair of polyominoes of size n, as a function of n, and the vertical bars show the ranges of the numbers. The data suggest that the average value of $\mathrm{NC}(n,n)$ for two random polyominoes grows linearly with n. With the available data for $3 \le n \le 14$ (considering the first two values as outliers), a linear regression gives the relation $\mathrm{NC}(n,n) \approx 2.19n + 4.97$.

Similar patterns of distributions of the number of compositions are observed also for polyominoes of different sizes. In order not to clutter the figure, we show overlays of distributions of the number of compositions of pairs of polyominoes of the same total size. Figures 10(c–d) show the distributions of the number of compositions of pairs of polyominoes whose total size is 12 and 14, respectively.

7 Conclusion

In this paper, we provide almost tight bounds on the minimum and maximum possible numbers of compositions of two polycubes in two and higher dimensions. While this goal is easy to achieve in three and higher dimensions, much more effort is needed in the two-dimensional case. We also provide an efficient algorithm for computing the number of compositions that two given polycubes have.

Future research directions include an estimation of the *average* number of composition two polyominoes have. An efficient upper bound on this number may overcome the error in Ref. [1] and yield an upper bound on the growth constant of polyominoes.

References

[1] G. Barequet and R. Barequet, An improved upper bound on the growth constant of polyominoes, *Proc. 8th European Conf. on Combinatorics, Graph Theory, and Applications*, Bergen, Norway, August-September 2015, *Electronic Notes in Discrete Mathematics*, 49, 67–172, November 2015, doi:10.1016/j.endm.2015.06.025.

[2] A. Asinowski, G. Barequet, G. Ben-Shachar, M.C. Osegueda, and G. Rote, On the number of compositions of two polycubes, In: *Proc. 11th European Conf. on Combinatorics, Graph Theory, and Applications*, eds. Jaroslav Nešetřil, Guillem Perarnau, Juanjo Rué, and Oriol Serra, Springer-Verlag, 2021, pp. 71–77, doi:10.1007/978-3-030-83823-2_12.

[3] G. Barequet, G. Rote, and M. Shalah, An improved upper bound on the growth constant of polyiamonds, *Proc. 10th European Conf. on Combinatorics, Graph Theory, and Applications*, Bratislava, Slovakia, *Acta Mathematica Universitatis Comenianae*, 88, 429–436, August 2019. `http://www.iam.fmph.uniba.sk/amuc/ojs/index.php/amuc/article/view/1205`

[4] S.R. Broadbent, J.M. Hammersley, Percolation processes: I. Crystals and mazes, *Proc. Cambridge Philosophical Society*, **53**, 629–641, 1957, doi:10.1017/S0305004100032680.

[5] S.W. Golomb, *Polyominoes*, Princeton University Press, Princeton, NJ, 2nd ed., 1994.

[6] L.H. Loomis and H. Whitney, An inequality related to the isoperimetric inequality, *Bull. of the American Mathematical Society*, **55**, 961–962, 1949, doi:10.1090/S0002-9904-1949-09320-5.

[7] C. Tricot, *Curves and Fractal Dimension*, Springer Science & Business Media, 1994.