The LEGO Counting Problem

Søren Eilers

Abstract. We detail the history of the problem of deciding how many ways one may combine n 2×4 LEGO bricks, and explain what is known—and not known—about the related question of how these numbers grow with n.

1. HISTORICAL BOUNDS. For decades, the LEGO Company (since 2005: The LEGO Group) would state in promotional material that six of the company's iconic 2×4 bricks could be combined in 102981500 ways if they had the same color. The author coincidentally became aware that this number was incorrect in 2003, and in 2004 computed the correct number which is almost 9 times larger. It is a key purpose of this note to explain how the correction was obtained, but let us first discuss the history of the problem as indeed this is highly instructive for understanding its solution.

With the help of the LEGO Group Archive, the number 102981500 has been traced back to 1974. It appeared in two short notes ([**6**],[**5**]) in the company's newsletter as an example of the use of the formula

$$t_n = 22 \sum_{i=0}^{n-2} 46^i \cdot 2^{n-2-i} + 2^{n-1}, \tag{1.1}$$

which was found by Jørgen Kirk Kristiansen, a chemical engineer working in the company labs who is also the grandson of the founder of the LEGO Group. Mr. Kristiansen was fully aware that he did not count all possible buildings and stated so explicitly in the note, explaining that building number 3 in Figure 1 is not counted, whereas buildings 1 and 2 are.

In fact, formula (1.1) gives a completely correct—but, as we shall see, unnecessarily complicated—description of the number of buildings of maximal height, counted in the sense we will describe below. Moreover, the values

$$t_2 = 24, \quad t_3 = 1060$$

were correctly computed this way. Precisely how and when this happened remains unclear, but over the course of the years it was forgotten that (1.1) was only intended as a **lower bound** of the number of buildings, and hence the number 102981500, which is $t_6 - 4$, was presented as the exact number of buildings in the LEGO Company's official communication, for instance in the 2004 company profile along with other "LEGO facts and figures" such as

> *It would take 40,000,000,000 LEGO bricks stacked on top of each other to reach from the Earth to the Moon*

http://dx.doi.org/10.4169/amer.math.monthly.123.5.415
MSC: Primary 05A16, Secondary 05B30

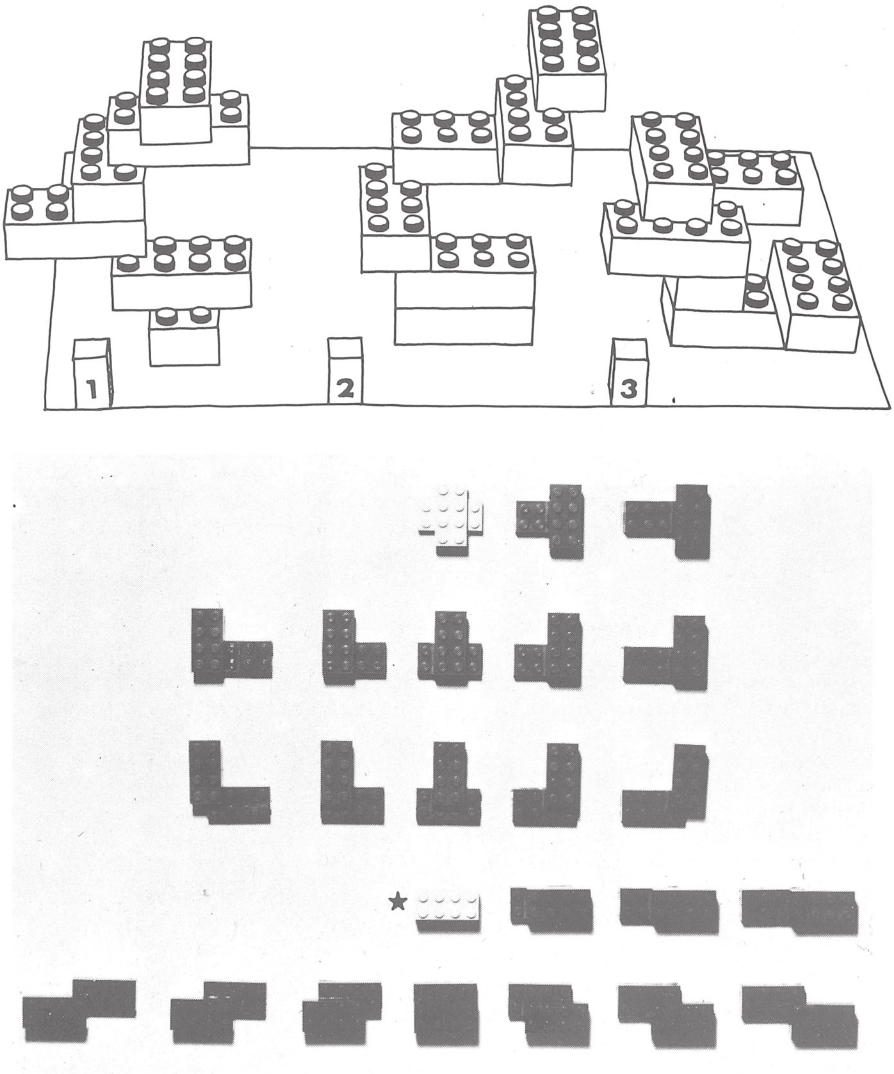

Figure 1. Illustrations from [**6**], [**5**]. (a) Three buildings. (b) $a_2 = 24$.

and

On average each person on Earth owns 52 LEGO bricks.

To avoid more misunderstandings, let us be very precise about what we are intending to count. We fix the dimensions $b \times w$ with $b \le w$ of a brick in the LEGO product range and count all buildings that are **contiguous**. By contiguous we mean that any brick B_0 is connected to any other brick B' in the sense that there is a number $\ell \ge 0$ and bricks $B_1, \dots, B_\ell$ so that B_0 is attached to B_1, B_1 is attached to B_2, etc., and B_ℓ is attached to B'. We only consider buildings in which all bricks are placed with top and bottom parallel to the XY-plane and with two of its sides parallel to the X-axis, and identify buildings which may be obtained from each other by translations in all of $\mathbb{R}^3$ or rotations in the XY-plane. Thus, in Figure 2 the configuration (a) is not counted, and the two configurations (b) and (c) are counted as one. We denote by $a_n^{b\times w}$ the number

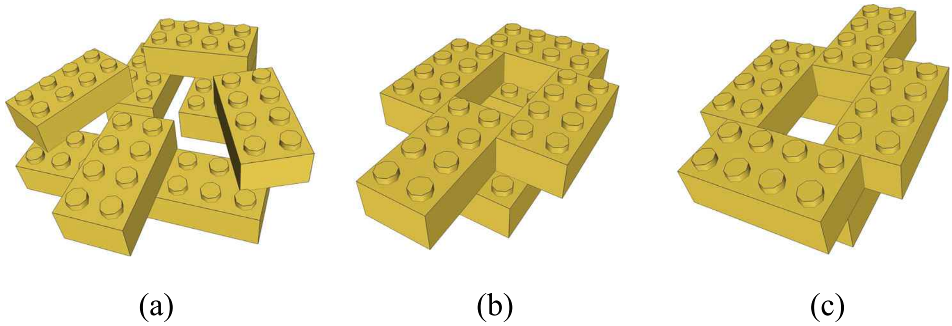

Figure 2. (a) Not counted. (b) and (c) Counted as one.

of such (equivalence classes of) buildings which can be obtained by n $b \times w$ bricks, and abbreviate $a_n = a_n^{2\times 4}$.

As we shall see below, apart from the decision to only consider buildings where all sides are parallel or perpendicular, it is of little mathematical consequence which conventions are used, but let us convince ourselves that we are using the same conventions as Mr. Kristiansen. Indeed, as observed in [**6**], there are 46 different ways to place one brick on top of another when the lower one is fixed, and 2 of these are self-symmetric after a rotation by 180°, whereas the remaining 44 buildings come in pairs defining 22 different buildings in our sense. Thus, as illustrated in Figure 1 (with the symmetric buildings colored white) there are exactly 24 different buildings with two bricks. Furthermore, it is now clear that when we fix one brick and place the remaining $n-1$ bricks on top of each other, we have a total of 46^{n-1} different choices for doing so. To obtain a building which is invariant under a rotation by 180°, we must choose one of the two exceptional configurations at every level, so 2^{n-1} of these choices lead to unique self-symmetric buildings, whereas the remaining choices come in pairs. In total, the number of buildings of height n become

$$t_n = \frac{1}{2}(46^{n-1} - 2^{n-1}) + 2^{n-1} = \frac{1}{2}(46^{n-1} + 2^{n-1}), \tag{1.2}$$

which is consistent with (1.1), since

$$\sum_{i=0}^{n-2} 46^i \cdot 2^{n-2-i} = 2^{n-2}\sum_{i=0}^{n-2} 23^i = 2^{n-2}\frac{23^{n-1}-1}{23-1}.$$

Let us digress a bit to note that the numbers 24, 1060, and 102981500 had in fact been a matter of contention at the LEGO Company in the early 1990's. When an exhibition "The Art of LEGO" was prepared at London's Science Museum, the head of the LEGO Company in the United Kingdom, Clive Nicholls, got interested in the problem and made the point that since any child would create 46 different configurations when asked to build all possible buildings with two bricks, LEGO should communicate the higher numbers 46, $46^2 = 2116$ and $46^5 = 205962976$ instead. Apart from seeing no reason to undersell LEGO's versatility, Mr. Nicholls had the further point that since any LEGO brick has the company logo printed in fine print inside each stud, it is in fact possible to distinguish a brick from its 180° rotation.

Mr. Nicholls got an answer from the very top of the organization ([**8**]), formulated by board member Per Sørensen:

The science of form is called morphology. It includes the concept of isomorphism—in this case, the ability of two or more objects to assume the same shape. All objects are isomorphous which by rotation in three-dimensional space and/or enlargement or reduction can be made to have the same shape. [...] An eight-stud LEGO element is isomorphous with an eight-stud DUPLO element, and white and red eight-stud bricks are also isomorphous with each other. [...] When the elements are isomorphous with each other, variations in which Element I is fitted on top of Element II are not morphologically different from variations in which Element II is fitted on top of Element I—even though, in purely physical terms they are of course different, because (whatever the people in the moulding shop may say) two elements are not the same. If the LEGO logo on the studs is turned one way or the other, this is—morphologically speaking—uninteresting, because it can be regarded as an unintentional difference and thus insignificant in terms of the morphological nature of the object.

Clive Nicholls had two words for that: "Morphology, schmorphology," and after a long tirade in the company newsletter, he threatened to leave the LEGO Company for the then archenemy TYCO, a subsidiary of Mattel, if the board did not revise the numbers. On a conciliatory note, Mr. Sørensen closed the discussion as follows: *I propose that in the future, we answer the question in the same way that Rolls Royce answers questions about horse power: enough!*

Before moving on to a discussion of how to find a_n by use of computers, note that we at least now have $a_2 = 24$ and the lower bound $a_n \geq t_n$ which tells us that a_n grows at least as fast as exponentially with base 46. In order to get any sort of theoretical handle on this problem, we need to complement this observation with an *upper bound* of the same nature. Finding such an upper bound would presumably be anathema to the communications division of LEGO Group, and since it is actually a good deal harder than providing a lower bound, it seems rather safe to assert that this was attempted for the first time by the author. Incidentally, the solution given in joint work with Durhuus ([**3**]) draws on another idea perfected by the LEGO Group: **The building instructions**.

To obtain a useful upper bound for a_n, valid for any n, we will use the approach that any building can be created by a set of instructions, and then count the possible instructions instead of the buildings themselves. To be able to implement such an overcounting strategy, however, we need to work with building instructions of a less immediate nature than what the average LEGO user would prefer. For $n \geq 2$ we will say that a map

$$\varphi : \{1, 2, \dots, 16n - 24\} \to \{-8, -7, \dots, 7, 8\}$$

is an *instruction* when $\varphi(i) \neq 0$ for precisely $n - 1$ values of i.

To use such an instruction, we enumerate the studs of the brick $1, \dots, 8$ starting in the top left corner and working left to right from the top row. First take one brick and call it brick 1. Then read $\varphi(1), \dots, \varphi(8)$ from left to right to specify what to build on top of brick 1 as follows. If $\varphi(1) > 0$, take another brick and place it parallel to brick 1 with hole $\varphi(1)$ on top of stud 1. If $\varphi(1) < 0$, take a brick and place it orthogonally, rotated $+90^\circ$, to brick 1 with hole $-\varphi(1)$ on top of hole 1. In both cases, give the new brick the number 2. If $\varphi(1) = 0$, do nothing. Then proceed to read $\varphi(2)$ to see what, if anything, to place on stud 2, and so on until $\varphi(8)$. Enumerate the bricks as they are introduced. When $n > 2$, similarly interpret $\varphi(9), \dots, \varphi(16)$ as an instruction of which bricks, if any, to place on top of brick 2, and $\varphi(17), \dots, \varphi(24)$ as instruction for what to place underneath brick 2, reading this time $\pm\varphi(i)$ as a specification of a stud to be placed in a hole, and continue this way to the end of the instruction.

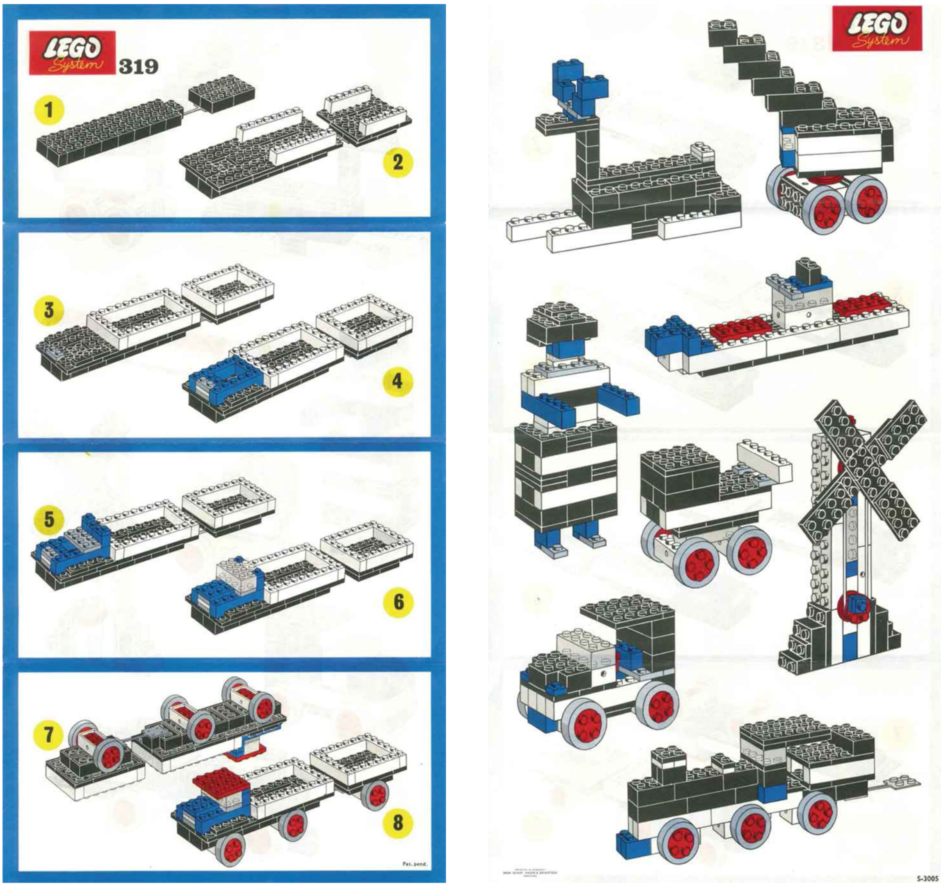

Figure 3. Building instructions, 1965. (Used with permission. ©2015 The LEGO Group.)

See Figures 4(a) and (b) for two examples of instructions defining buildings we are attempting to count. We note, however, that two things can go wrong when one attempts to follow such instructions—as in Figure 4(c) the bricks specified can collide, and as in Figure 4(d) we may encounter a situation where no brick number $\ell + 1$ has been introduced when we have reached the end of the specifications of what to place on bricks $1, \ldots, \ell$. We also note that in most situations, there are many different instructions leading to the same building.

But since, just as is the case for LEGO Group building instructions (cf. Figure 3), there is obviously an instruction which will create any given building among the ones we are aspiring to count, the number of instructions is larger than the number of buildings. And we can count the instructions as

$$\binom{16n-24}{n-1}16^{n-1},$$

since the binomial coefficient enumerates the number of possible positions of nonzero values and 16^{n-1} enumerates the number of choices for the nonzero entries. To avoid the computational complexity of computing binomial coefficients, appealing to Stirling's formula one can see that these numbers grow no faster than $(16^{17}/15^{15})^{n-1} \simeq 674.02^{n-1}$. In fact, we will always have that the number of instructions, and hence -

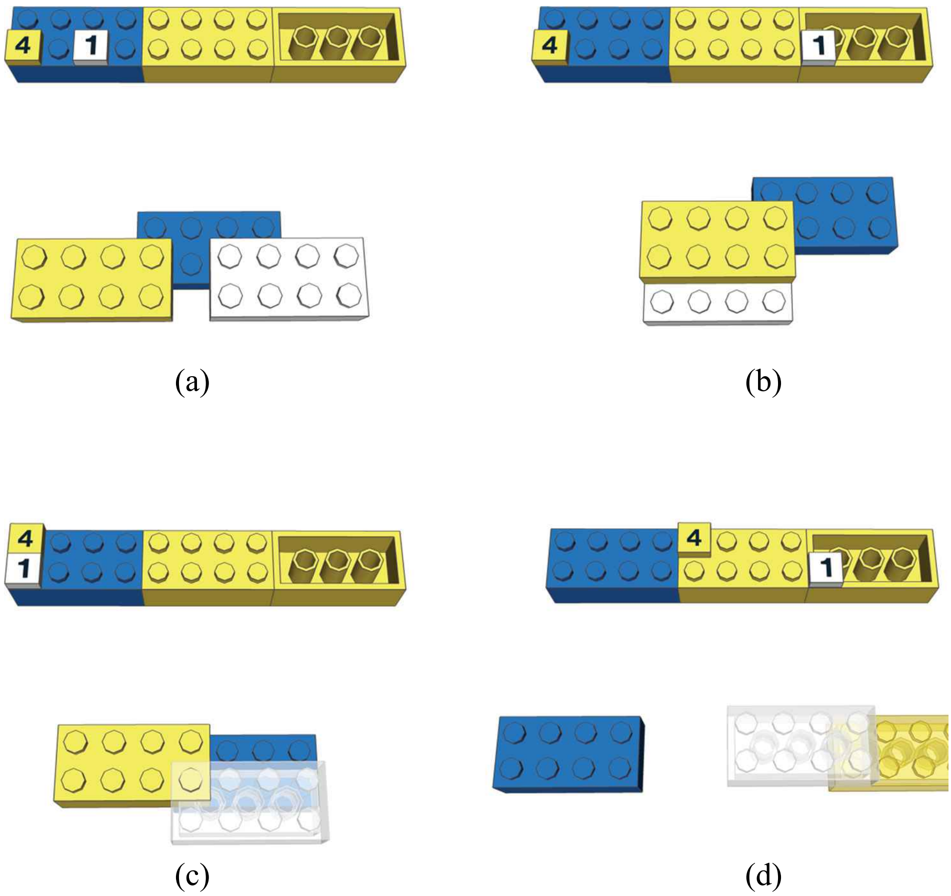

Figure 4. Instructions and resulting buildings

a_n, - is bounded by $u_n = 675^{n-1}$. Thus a_n grows at most as fast as exponentially with base 675.

In conclusion, we may now say with certainty that the number of ways to combine six 2×4 bricks lies somewhere between 102981504 and $675^5 = 140126044921875$. To narrow it down we need to use a computer.

2. COUNTING WITH COMPUTERS. In 2011, the author was made aware that he was not the first to try to remedy that the numbers provided by LEGO Company were only covering a subset of all buildings. On the LEGO user group's electronic discussion forum LUGnet, a user already in 2002 posted the argument leading to (1.2), and added:

> *There remains the problem for the case where the solid is not necessarily an n-story building. I only have a result 1560 for $n = 3$ using a computer. I think it is computable until $n = 5$ or 6.*

n	a_n	
1	1	
2	24	Kristiansen 1974
3	1560	Anonymous 2002
4	119580	Eilers 2004
5	10166403	Eilers 2004
6	915103765	Eilers 2004
7	85747377755	Abrahamsen-Eilers 2006
8	8274075616387	Abrahamsen-Eilers 2006
9	816630819554486	Nilsson 2012

Figure 5. Known values of a_n (A112389 of [**7**])

As indicated in Figure 5, the prediction on how far a_n was computable was a bit on the pessimistic side, but the claim that $a_3 = 1560$ is correct. And the anonymous[1] LEGO enthusiast was certainly hitting the nail on the head by predicting that issues concerning *efficiency of computation* would come up.

We have already touched upon such issues; indeed the main reason for finding our revised formula (1.2) superior to the original (1.1) is that it is faster to compute, and apart from our desire to provide an upper bound of the same nature as the lower bound, we emphasized the feature that 675^{n-1} was efficiently computable also for large n. Indeed, even though the numbers t_n and u_n grow quickly as n increases, we may use logarithms, successive squaring, or other standard computational methods, to compute the numbers at an expense in time which grows at worst as a linear expression in n, or, - which is the same, - as a linear expression in the number of digits in the computed numbers.

But when it comes to computing a_n, we do not know of any method which does not require us to go through, one at a time, a large part of the possible configurations, and since the number of buildings grows exponentially with n, so does the time consumption. The first attempt by the author, naively going through all possible buildings saving time only by employing our conventions of identification, could compute up to $a_6 = 915103765$. That number, which the LEGO Group in short order accepted and helped disseminate widely, was the main goal of the initial investigations, but computing it required almost a week's computing time on a laptop, and hence there was little hope to compute beyond $n = 6$.

The situation was improved somewhat in joint work with Abrahamsen ([**2**]), who among many other things made the observation that it is more efficient to count closer to Mr. Nicholls' convention, fixing a brick and then counting all buildings containing this brick at its base level. One issue, then, that perhaps Mr. Nicholls had not considered, is that when the base level contains more than one brick arranged with its long side parallel to the base brick, say k such bricks, then every configuration will be counted k times even if one does not allow identifications by rotations, but only by translations. But taking this into account, and keeping track also of which buildings are symmetric after rotations, we may compute a_n by

$$a_n = \sum_{m=1}^{n} \frac{c(n,m) + c^{180}(n,m) + 2c^{90}(n,m)}{2m},$$

[1]The author of the post has been identified, but prefers to remain anonymous.

where $c(n, m)$ is the number of configurations with n bricks containing the base brick in its bottom level, so that there are m bricks in the lower level, and where $c^{180}(n, m)$ and $c^{90}(n, m)$ count those that are symmetric after rotations by 180 and 90 degrees in the XY-plane as indicated.

However, it is obviously unnecessarily inefficient to work our way through all buildings this way, since (1.1) allows us to quickly count all the buildings of maximal height. Let's elaborate on the idea implicit in Mr. Kristiansen's computation. Whenever we know that there is only one single brick in some layer of the building, we can compute the number of possibilities by multiplication of the number of possibilities of what to put below and the number of possibilities of what to put on top. We may speed up the computations substantially by defining $\mathbf{c}(n, m)$ as the number of buildings with n bricks, m of which are in the bottommost level, which are **fat** in the sense that at every level above the bottommost, there are at least two bricks. Also, define $\overline{\mathbf{c}}(n)$ as the number of buildings with $n + 1$ bricks so that there is one brick each in the topmost and bottommost level, and two or more in any other level. For instance, building 3 in Figure 1 is one of the buildings counted by $\overline{\mathbf{c}}(5)$. It is elementary, but tedious, to verify then that

$$\begin{aligned} a_n = &\sum_{m=2}^{n} \frac{\mathbf{c}(n, m) + \mathbf{c}^{180}(n, m) + 2\mathbf{c}^{90}(n, m)}{2m} \\ &+ \frac{1}{2} \sum_{\ell=0}^{n-1} \sum_{m_1+m_2+k_1+\cdots+k_\ell=n+1} \big[\mathbf{c}(m_1, 1)\mathbf{c}(m_2, 1)\overline{\mathbf{c}}(k_1) \cdots \overline{\mathbf{c}}(k_\ell) \\ &\qquad + \mathbf{c}^{180}(m_1, 1)\mathbf{c}^{180}(m_2, 1)\overline{\mathbf{c}}^{180}(k_1) \cdots \overline{\mathbf{c}}^{180}(k_\ell)\big]. \end{aligned} \tag{2.3}$$

Formula (2.3) looks rather formidable, but has several mitigating features. First, we note that since the 2×4 brick is not itself invariant under a rotation by 90°, it takes at least 4 bricks (two with the long side parallel to the X-axis, two with the long side parallel to the Y-axis) to create a layer which is invariant under such a rotation, and hence $\mathbf{c}^{90}(n, m) = 0$ unless 4 divides both m and n—the first nonzero value is $\mathbf{c}^{90}(8, 4) = 244$. Second, since $\mathbf{c}(n, m) = 0$ when $n \leq m + 2$ (unless $n = m = 1$) and since $\overline{\mathbf{c}}(2) = 0$, the expressions reduce substantially for small n. Indeed, we rediscover

$$a_2 = \frac{1}{2}\left(\overline{\mathbf{c}}(1) + \overline{\mathbf{c}}^{180}(1)\right) = t_2$$

and find

$$\begin{aligned} a_3 &= \frac{1}{2}\left(\overline{\mathbf{c}}(1)^2 + 2\mathbf{c}(3, 1) + \overline{\mathbf{c}}^{180}(1)^2 + 2\mathbf{c}^{180}(3, 1)\right), \\ a_4 &= \frac{1}{4}\left(\mathbf{c}(4, 2) + \mathbf{c}^{180}(4, 2)\right) + \frac{1}{2}\big(\overline{\mathbf{c}}(1)^3 + 2\mathbf{c}(3, 1)\overline{\mathbf{c}}(1) + \overline{\mathbf{c}}(3) + 2\mathbf{c}(4, 1) \\ &\quad + \overline{\mathbf{c}}^{180}(1)^3 + 2\mathbf{c}^{180}(3, 1)\overline{\mathbf{c}}^{180}(1) + \overline{\mathbf{c}}^{180}(3) + 2\mathbf{c}^{180}(4, 1)\big), \end{aligned}$$

where all of the necessary constants are listed in Figure 7.

The point of (2.3) is of course that the number of fat buildings grows slower than the total number of buildings. This helps, but it doesn't help a whole lot, since to this day we know of no way to avoid more or less individually counting the fat buildings. Thus, the time needed to compute a_n is at least as large as a number proportional to

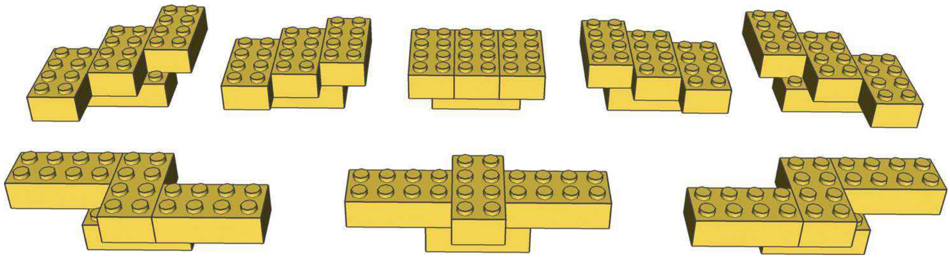

Figure 6. $\mathbf{c}^{180}(4, 1) = 8$

n	$\overline{\mathbf{c}}(n)$	$\overline{\mathbf{c}}^{180}(n)$	$\mathbf{c}(n,1)$	$\mathbf{c}^{180}(n,1)$	$\mathbf{c}(n,2)$	$\mathbf{c}^{180}(n,2)$
1	46	2	1	1	0	0
2	0	0	0	0	0	0
3	74130	32	480	20	0	0
4	867346	24	1288	8	41514	130

Figure 7. Some basic counts

$\overline{\mathbf{c}}(n-1)$, and these numbers can be proven (as we will see below) to grow at least exponentially with base $\sqrt{1248} \simeq 35.3$. Thus, unless a better way is found to count fat buildings, the computation time needed to compute a_n will grow exponentially with a prohibitively large base. The author does not believe it can be done in polynomial time, but has no formal evidence for such a claim.

The concrete programs used in [**2**] (see [**1**] for more details) could compute a_6 in about 5 minutes, but with an increase in computing times of around 100 for each additional brick, finding a_8 took about 500 CPU hours and finding a_9 was projected to take more than 5 CPU years. Thus the author was rather awed when approached in 2012 by Johan Nilsson, a Swedish mathematician then based in Germany, who could not only supply a_9 but had also independently verified $a_1, \ldots, a_8$.

Dr. Nilsson's approach was to *parallelize* the problem. The algorithms used in the author's computations do not lend themselves well to such an approach, but Nilsson had the brilliant idea of running through all instructions instead, one at a time, checking which gave rise to buildings. Dividing the universe of instructions evenly among a large number of computers at the Department of Mathematics at the University of Bielefeld, which were working on the problem when otherwise idle, Nilsson could obtain a_9 in a matter of months.

3. THE GROWTH CONSTANT. The most efficient way of communicating the versatility of the 2×4 brick, rather than a sequence of individual counts, would be via the **growth constant** h defined so that

$$a_n \approx k \cdot h^n.$$

Such growth constants are ubiquitous in asymptotic combinatorics and are key concepts in applications, measuring **capacity** in contexts of information theory or computer science, or **entropy** in contexts of physics.

That such a constant h is defined is nontrivial and requires some interpretation of what we mean by "$\approx$." Of course if we knew that a_{n+1}/a_n converged as $n \to \infty$, the limit would be an excellent candidate, but although this is in all likelihood the case, the author knows of no way of proving it. Instead, which is nearly as useful, we may use our upper bound to prove that $\sqrt[n]{a_n}$ converges by the following lemma.

Lemma 3.1. *[3]* $\lim_{n\to\infty} \log a_n/n$ *exists.*

Proof: We note first that

$$\lim_{n\longrightarrow\infty} \frac{\log a_n}{n} = \lim_{n\longrightarrow\infty} \frac{\log c(n,1)}{n} \tag{3.4}$$

in the sense that if one limit exists, so does the other. Note that since we have found that $\log a_n \le \log u_n = (n-1)\log 675$, the sequence $\log a_n/n$ is bounded. Our claim (3.4) follows immediately by the inequalities

$$a_{n-1} \le c(n,1) \le 2a_n.$$

The leftmost inequality follows by mapping each equivalence class of configurations with $n-1$ bricks to a representative placed on top of a fixed base brick and noting that this map is injective. The rightmost follows by mapping each configuration to an equivalence class and noting that this map is at most $2-1$.

Letting $\mathcal{C}_n$ denote the set of buildings counted by $c(n,1)$, one sees that $c(n+m,1) \ge c(n,1)c(m,1)$ by noting that an injective map from $\mathcal{C}_n \times \mathcal{C}_m$ to $\mathcal{C}_{m+n}$ is defined by placing the base brick of the element of $\mathcal{C}_m$ somewhere on the top layer of the element of $\mathcal{C}_n$. Hence, $\log c(n,1)$ is a superadditive sequence, and appealing to Fekete's lemma, $\log c(n,1)/n$ converges to $\sup_{n\in\mathbb{N}} \log c(n,1)/n$ in $[0,\infty]$. But we have seen that the limit is finite; indeed it is less than $\log 675$. $\square$

Taking exponentials, we set

$$h = \lim_{n\to\infty} \sqrt[n]{a_n} = \lim_{n\to\infty} \sqrt[n]{c(n,1)},$$

noting in particular that when we focus on h rather than individual counts, Mr. Nicholls' protests become completely inconsequential. Convincing ourselves that upper bounds $u_n^{b\times w} = (s^{b\times w})^n$ for any dimensions can be obtained by counting instructions, we see further that

$$h^{b\times w} = \lim_{n\to\infty} \sqrt[n]{a_n^{b\times w}}$$

makes sense for any choice of dimension $b \times w$. Thus, we have an extremely efficient measure of the versatility of each brick in the LEGO product line, which can meaningfully be compared among themselves, and to other such measures. But before we can ask the LEGO Group to start saying something like

> *Already have a lot of* 2×4 *LEGO bricks? Buy one more and have the number of buildings you can create multiplied by* h*!*

we have to face up to the task of computing, or at least estimating, such numbers h.

Our lower and upper bounds tell us that $46 \le h \le 675$, leaving a lot of room for improvement. The first step is to scrutinize our definition of instructions with the aim of reducing the upper bound. For instance, since 38 of the 46 ways to place one brick on top of another involves more than one stud, we can avoid some redundance by distributing the positions evenly with at most 6 choices for each stud. Furthermore, one can use that one stud (or one hole) has already been spoken for when placing all bricks except the first to reduce the number of possible choices on the side which has already been in use from 46 to 30. One checks that 30 positions can be distributed evenly on 5 studs, leading to

$$\binom{(8+5)n-(8+5+5)}{n-1}6^{n-1}$$

instructions, and the ensuing estimate

$$h \le (13^{13}/12^{12}) \cdot 6 < 204.$$

Adapting much more advanced methods developed in the context of enumerating polyominoes ([**4**]), it was proved in [**3**] that $h < 177$.

The lower bound can be improved somewhat by appealing to the concept of **generating functions**. Organizing the individual counts into a power series

$$A(z) = \sum_{n=1}^{\infty} a_n z^n = 46z + 1560z^2 + 119580z^3 + \cdots$$

we see by the root criterion that the sum converges in $[0, 1/h)$ and diverges in $(1/h, \infty)$. Using standard methods from the theory of generating functions, (2.3) translates to

$$A(z) = \sum_{m=2}^{\infty} \frac{C_m(z) + C_m^{180}(z) + 2C_m^{90}(z)}{m} + \frac{C_1(z)^2}{2z(1-\overline{C}(z))} + \frac{C_1^{180}(z)^2}{2z(1-\overline{C}^{180}(z))}$$

with functions $C_m, C_m^{180}, C_m^{90}, \overline{C}, \overline{C}^{180}$ defined from constants $\mathbf{c}(n,m)$, $\mathbf{c}^{180}(n,m)$, $\mathbf{c}^{90}(n,m)$, $\overline{\mathbf{c}}(n)$, and $\overline{\mathbf{c}}^{180}(n)$ in the same way that we defined A from a_n. Moreover, since all of these functions must converge on $[0, 1/h)$, since $\mathbf{c}(n,m)$, $\mathbf{c}^{180}(n,m)$, $\mathbf{c}^{90}(n,m) < a_n$, and since $\overline{\mathbf{c}}(n), \overline{\mathbf{c}}^{180}(n) \le c(n,1)$, we may conclude (after a little more work) that $A(z)$ diverges at $1/h$ as a result of division by zero:

$$\overline{C}(1/h) = 1. \tag{3.5}$$

More precisely, h is the reciprocal of the smallest solution to $\overline{C}(x) = 1$ on $[0, 1]$. We know, as recorded in Figure 7,

$$\overline{C}(x) \ge 46x + 74130x^3 + 867346x^4$$

on $[0, 1]$, so solving for x we obtain that $h > 66$. Using more values of $\overline{\mathbf{c}}(n)$, the last being

$$\overline{\mathbf{c}}(9) = 2067477693115$$

as computed by Nilsson, and the general estimate $\overline{\mathbf{c}}(n+2) > 1248\overline{\mathbf{c}}(n)$, we may show (as in [**3**]) that $h > 81$.

But it remains a sad fact, not for want of trying, that this is the best the author has been able to do, and hence the ambitions for using growth constants to gauge and compare the versatility of different brick sizes is largely unrealized. For instance, we can create upper bounds by counting instructions to see that both $h^{1\times 2}$ and $h^{2\times 2}$ are less than 81, and hence prove the nonsurprising fact that the 2×4 brick is more versatile than both the 1×2 and the 2×2 brick. But because of overlaps between the intervals in which we know that $h^{1\times 2}$ and $h^{2\times 2}$ must be contained, we are not able to say with certainty which of these bricks is more versatile. By comparing $a_n^{1\times 2}$

$$1, 4, 37, 375, 4493, 56848, 753536, 10283622, 143607345$$

to $a_n^{2\times 2}$

$$1, 3, 31, 412, 6435, 106108, 1825803, 32320892, 584956651$$

for $n \in \{1, \dots, 9\}$ it appears that the 2×2 brick is superior.

Although it felt a bit like acknowledging defeat, we in [**3**] took to heuristic estimation of h by the standard method of fitting a straight line to a semilogarithmic plot. The best fit to our observations $a_1, \dots a_8$, however, gave the value $\widehat{h} \approx 74.8$ which was not consistent with our lower bounds, indicating that we had too few observations for such an approach. In [**2**] we consequently applied Monte Carlo methods, estimating a_n by drawing instructions at random, seeing how often they gave rise to actual buildings to estimate $a_9, \dots, a_{20}$, to arrive at $\widehat{h} \approx 117$.

This remains the author's best guess, but of course it should be taken only for what it is: a guess. For instance, we now know that our estimate $a_9 \approx 7.94 \times 10^{14}$, obtained 5 years before Nilsson provided the exact value, was almost 3% too low. Imprecisions of this nature can be expected to cancel out, but this leaves the real problem that we have no way of knowing how well the growth of $a_1 \dots, a_{20}$ predicts the true value of h.

ACKNOWLEDGMENT. The author was supported by VILLUM FONDEN through the network for Experimental Mathematics in Number Theory, Operator Algebras, and Topology, as well as the Danish National Research Foundation through the Centre for Symmetry and Deformation (DNRF92). The author further wishes to thank records manager Tine Froberg Mortensen at the LEGO Group Archives for her invaluable assistance.

REFERENCES

1. M. Abrahamsen, S. Eilers, Efficient counting of LEGO structures, Tech. Report www.math.ku.dk/~eilers/eclbii.pdf, University of Copenhagen, 2007.
2. ———, On the asymptotic enumeration of LEGO structures, *Exp. Math.* **20** (2011) 145–152.
3. B. Durhuus, S. Eilers, On the entropy of LEGO, *J. Appl. Math. Comput.* **45** (2014) 433–448.
4. D. A. Klarner, R. L. Rivest, A procedure for improving the upper bound for the number of n-ominoes, *Canad. J. Math.* **25** (1973) 585–602.
5. J. K. Kristiansen, Mere taljonglering med klodser, *Klodshans* **3** (1974) 13 [Danish].
6. ———, Taljonglering med klodser – eller talrige klodser, *Klodshans* **2** (1974) 12 [Danish].
7. N. J. A. Sloane, The on-line encyclopedia of integer sequences, http://oeis.org.
8. K. Sørensen, Morphology in practice, *LEGO Rev.* **2** (1991) 8.

SØREN EILERS obtained his Ph.D. from the University of Copenhagen in 1995. He is currently on sabbatical from his position there, acting as the main organizer of the program "Classification of operator algebras: Complexity, rigidity, and dynamics" at Institut Mittag-Leffler in Stockholm. After being featured as the crazy mathematician in "A LEGO Brickumentary," his Bacon–Erdős number dropped from ∞ to 7.
Department of Mathematical Sciences, University of Copenhagen, Universitetsparken 5
DK-2100 Copenhagen Ø, Denmark
eilers@math.ku.dk