ENTROPY OF LEGO® JUMPER PLATES

DAVID M. MCCLENDON AND JONATHON WILSON

ABSTRACT. We study the growth rate of the function $T_{\mathcal{J}}(n)$ which counts the number of contiguous buildings that can be made from n LEGO "jumper plates" of the same color. We give upper and lower bounds on the exponential growth rate of $T_{\mathcal{J}}(n)$: the lower bound comes from techniques introduced by Durhuus and Eilers in [DE2] and the upper bound is derived by associating to each building a "labelled binary tree" and counting the number of such objects.

1. Introduction

1.1. **Background.** In 2016, revenue of the LEGO Company was more than \$6.3 billion [Le]. One reason LEGO products are so popular might be the seemingly endless number of ways to connect together the small plastic building toys. This leads to an interesting combinatorial question: exactly how many different ways can n LEGO bricks of the same size, color and shape be interlocked? If n is small, then this number can be counted exactly, if one has enough computing power. Begfinnur Durhuus and Søren Eilers studied this question for 2×4 rectangular LEGO bricks and were able to determine that there are

$$8,274,075,616,387$$

different ways to connect eight 2×4 LEGO bricks [DE2]. To put this number into perspective, suppose that you could build one of these constructions every five seconds. It would take you 1.31 million years to run through all these constructions!

Unfortunately, once n becomes large (for 2×4 bricks, "large" means 10 [DE2]), the exact number of different configurations is still not known - no closed mathematical formula exists, and the run time for any known computer algorithm is too large. The good news, however, is that if one defines $T_{\mathcal{B}}(n)$ to be the number of different configurations that can be built from n LEGO bricks from some particular class $\mathcal{B}$ of brick, then in many cases one can show that $T_{\mathcal{B}}(n)$ grows exponentially in n, and upper and lower bounds on the exponential growth rate of this function can be obtained. Indeed, in [DE2], Durhuus and Eilers compute upper and lower bounds on this growth rate for standard rectangular LEGO bricks.

In this paper, we study a different type of LEGO brick, called a **jumper plate**. A jumper plate is a 1×2 LEGO element which has only one stud on its top, and

---

Date: February 19, 2018.

2010 *Mathematics Subject Classification.* Primary 05A16, Secondary 05B30, 05C30.

Key words and phrases. LEGO, asymptotic enumeration, entropy, labelled binary tree.

LEGO is a trademark of LEGO Company.

The authors thank the organizers of the 2017 MOVES conference, especially Tim Chartier, who recommended that the second author give a talk on related work at the conference. We also thank Søren Eilers for providing computer computations of some constants in this paper, suggesting the application of our technique to roof tiles, and for other helpful discussions.

Figure 1. The bottom and top of a LEGO jumper plate. To attach two jumper plates, the stud on the top of the child can be inserted into any of the three "slots" on the bottom of the parent.

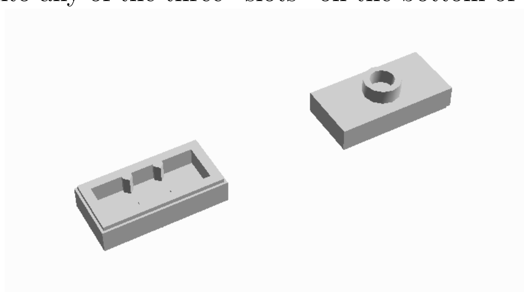
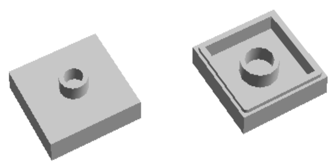


Figure 2. Top and bottom view of a LEGO double jumper plate.



three locations on its bottom into which studs can be inserted (see Figure 1). When two jumper plates are attached in this way, we can arrange them so that their studs point upward, and refer to the plate on the top of the connection as the **parent** and the plate underneath as a **child** (we use the term **grandchild** in the obvious way).

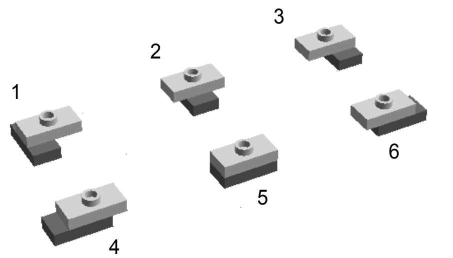
In this paper, we study the function $T_{\mathcal{J}}(n)$ which counts the number of contiguous LEGO buildings that can be made from n jumper plates. We are especially interested in determining how fast this function grows: is it exponential? super-exponential? etc. We will prove that $T_{\mathcal{J}}$ grows at an exponential rate, and give bounds on the rate of its exponential growth.

1.2. **Why jumper plates?** For some classes $\mathcal{B}$ of LEGO bricks, $T_{\mathcal{B}}(n)$ is very easy to figure. For example, for a standard 1×1 LEGO brick, the only way to connect n such bricks together is to make a 1×1 tower of height n, so $T_{1\times 1}(n)$ is the constant function $T_{1\times 1}(n) = 1$.

LEGO also produces a "double jumper" plate (denote this class of plate by $\mathcal{D}$) which is a 2×2 plate with a single stud in the center of the top (see Figure 2). There are 5 ways to attach one double jumper plate to another (by placing the stud of the child in either the center or one of the four corners of the parent). Since a double jumper plate can have at most one child, saying which of the 5 connections is used to attach each child to its parent completely describes a building made from n double jumper plates. Since there are $n-1$ plates in such a building which are children, $T_{\mathcal{D}}(n) = 5^{n-1}$.

One reason why these two classes of bricks have easy to describe functions T is that the buildings one can make from them lack three "dimensions" of freedom, in that the number of pieces being used completely determines the building's

FIGURE 3. The six different ways to attach two jumper plates. In each building, the light gray plate is the parent and the dark gray plate is the child. In the top three connections pictured, we say the child is perpendicular to the parent; in the bottom three connections, we say the child is parallel to the parent.



height. Jumper plates are the simplest LEGO elements that allow for bonafide three-dimensional constructions, in which one can build outwards in non-trivial ways as well as directly up and down, and that is why we choose to study them.

Jumper plates are popular with LEGO aficionados because unlike standard rectangular LEGO bricks, jumper plates allow for creations that have a "half-stud" offset; this "jumping" of a half-stud gives the piece its name.

1.3. **What makes two buildings "different"?** We said earlier that $T_{\mathcal{J}}(n)$ is the number of contiguous LEGO buildings that can be made from n jumper plates. To clarify this definition, we need to describe exactly what makes one building "different" from another. First, since each jumper plate has only one stud on its top, the building has to have a unique jumper plate on its top-most level; call this jumper plate the **root** of the building. To account for translational symmetry, we specify that $T_{\mathcal{J}}(n)$ is the number of buildings that can be made from n jumper plates, where the root occupies a fixed position.

If one thinks of buildings as being identified up to rotational symmetry, then each building is counted twice in our computation of $T_{\mathcal{J}}(n)$ (because when the root is rotated by 180° about its center, it occupies the same position). However, the exponential growth rate of $T_{\mathcal{J}}(n)$ would be the same whether such buildings are identified or not, so we will not bother with identifying buildings which are rotationally symmetric. As an example, in Figure 3 we treat buildings 1 and 3 as two separate buildings (each made from 2 jumper plates), even though by rotating building 1 by 180° produces building 3. In particular, this means $T_{\mathcal{J}}(2) = 6$. Notice that the "half-stud" offset permitted in some of the attachments shown in Figure 3 means that jumper plates will not form the same kinds of buildings as the standard rectangular LEGO bricks studied in [DE2].

1.4. **Configurations made from a small number of jumper plates.** To get an idea of how the function $T_{\mathcal{J}}$ behaves, let's actually compute some values of $T_{\mathcal{J}}(n)$. When $n = 3$, we can just build each of the constructions and count them (see Figure 4), and if n is small enough, we can count $T_{\mathcal{J}}(n)$ by hand (see Figure 5 for the values when $n \leq 8$). To get an idea of how these values are obtained, we'll go through the case $n = 5$. Buildings made from 5 jumper plates must have height

FIGURE 4. The 37 buildings that can be made from 3 jumper plates. Notice that the top-most plate in each construction (the root) occupies a fixed position. The bottom-most building in this picture is the only building of height 2 that can be made from 3 jumper plates; the other buildings all have height 3. These buildings of height 3 can be catalogued by first choosing one of the 6 connections described in Figure 3 to specify how to attach the middle plate to the root, and then choosing one of the 6 connections of Figure 3 for how the bottom plate attaches to the middle plate. This gives $6(6) = 36$ buildings of height 3 made from 3 jumper plates (in general, there are 6^{n-1} buildings of height n made from n jumper plates).

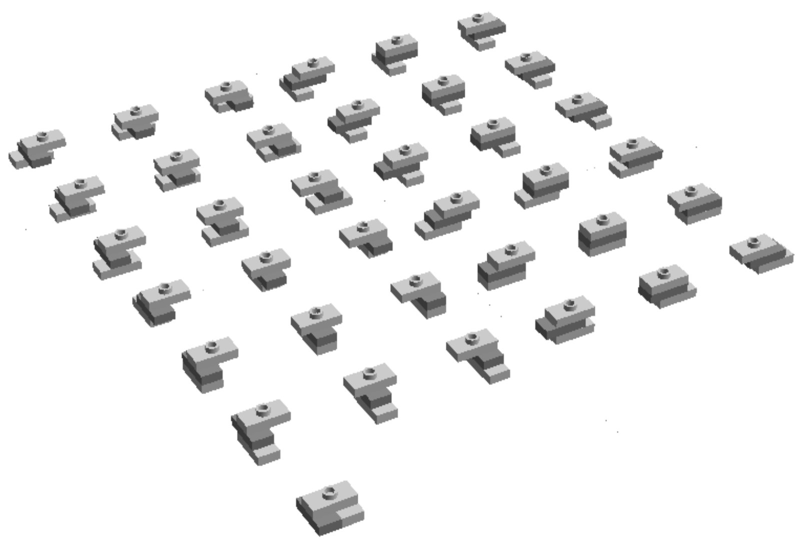


FIGURE 5. Values of $T_{\mathcal{J}}(n)$ for $n \leq 8$, computed by hand.

n	$T_{\mathcal{J}}(n)$
1	1
2	6
3	37
4	234
5	1489
6	9534
7	61169
8	393314

3, 4 or 5, so we can count the number of buildings of each height separately and add:

FIGURE 6. Values of $T_{\mathcal{J}}(n)$ for $9 \leq n \leq 14$, computed via computer calculations by S. Eilers [E]:

n	$T_{\mathcal{J}}(n)$
9	2531777
10	16316262
11	105237737
12	679336650
13	4388301841
14	28366361206

If the building is five plates high: here, every plate (other than the bottom one) has exactly one child; since there are 6 ways to attach each non-root plate to its parent, we obtain a total of $6^4 = 1296$ such buildings.

If the building has four plates high: exactly one of the five plates must have two children.

- If the root has two children, then the grandchild of the root must be attached to one of the 2 children in one of 6 ways (so there are $2 \cdot 6 = 12$ ways to attach the grandchild to the bottom of the building); then there are 6 ways to attach the last plate underneath the grandchild. So there are $12 \cdot 6 = 72$ buildings where only the root has two children.
- If the child of the root is the only plate with two children, then a similar argument yields 72 buildings in this case as well.
- If the grandchild of the root is the plate with two children, then there are 36 buildings (6 ways to attach the child to the root, 6 ways to attach the grandchild of the root underneath the child, and 1 way to attach the last two plates under the grandchild of the root).

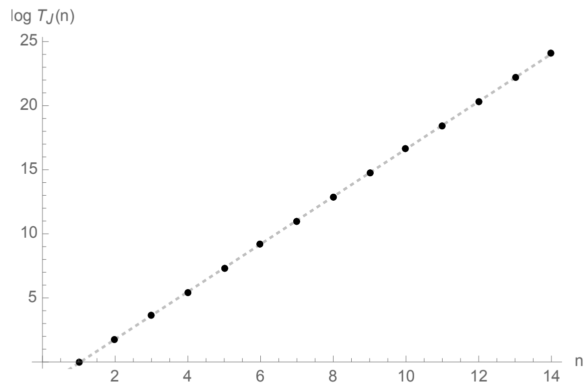
If the building is three plates high: the building consists of the root, the two children of the root, and two plates attached under the children of the root. The only freedom in such a building is in attaching the bottom-most plates; there are 13 ways to do this (9 ways in which the plates in the bottom level are parallel to the plates in the second level, and 4 ways in which the plates in the bottom level are perpendicular to those in the middle level).

Combining these cases, we obtain $T_{\mathcal{J}}(5) = 1296 + 72 + 72 + 36 + 13 = 1489$.

Sadly, once $n \geq 9$, this method begins to break down because there are too many cases to efficiently count. However, S. Eilers [E] recently communicated to us the values of $T_{\mathcal{J}}(n)$ for $9 \leq n \leq 14$, obtained by computer computations (see Figure 6); unfortunately, even computer computations do not help in computing $T_{\mathcal{J}}(n)$ for larger n–the run time for known algorithms becomes prohibitively large.

Plotting the values of $\log T_{\mathcal{J}}(n)$ against n for $n \leq 14$ (Figure 7), we see that $T_{\mathcal{J}}$ appears to have exponential growth (in fact, in Section 2 we will prove that $T_{\mathcal{J}}(n)$ does in fact grow exponentially). Furthermore, the least-squares linear equation for $\log T_{\mathcal{J}}(n)$ against n, derived from the values of $T_{\mathcal{J}}(n)$ for $n \leq 14$, suggests that $T_{\mathcal{J}}(n) \approx e^{1.85531n-1.93902}$, i.e. that $T_{\mathcal{J}}(n)$ has exponential growth rate $1.85531 \approx \log 6.39368$. (We will show rigorously in Section 3 that this numerical approximation underestimates, at least slightly, the actual exponential growth rate of $T_{\mathcal{J}}$.)

FIGURE 7. The graph of $\log T_{\mathcal{J}}(n)$ versus n for $n \in \{1, ..., 14\}$. The least-squares line derived from these points (shown by the dashed line) has equation $y \approx 1.85531x - 1.93902$, suggesting that $T_{\mathcal{J}}(n) \approx e^{1.85531n - 1.93902}$.



1.5. **Entropy.** We saw from the numerics in the previous section that there is good reason to believe $T_{\mathcal{J}}(n)$ grows exponentially, i.e. that $T_{\mathcal{J}}(n)$ grows like $C \exp(h_{\mathcal{J}} n)$ for suitable constants C and $h_{\mathcal{J}}$. We are interested in studying the value of $h_{\mathcal{J}}$; assuming $T_{\mathcal{J}}(n) \approx C \exp(h_{\mathcal{J}} n)$, we can "solve" for $h_{\mathcal{J}}$ to obtain

$$h_{\mathcal{J}} \approx \frac{1}{n} \left(\log T_{\mathcal{J}}(n) - \log C \right).$$

As $n \to \infty$, the $\frac{1}{n} \log C$ term goes to zero, leaving $h_{\mathcal{J}} \approx \frac{1}{n} \log T_{\mathcal{J}}(n)$. With this idea in mind, we define the **entropy** of a jumper plate to be

$$h_{\mathcal{J}} := \lim_{n \to \infty} \frac{1}{n} \log T_{\mathcal{J}}(n),$$

provided this limit exists (we will show that it does in Section 2).

We use the word "entropy" because the formula used to define $h_{\mathcal{J}}$ resembles a formula used in information science to compute a quantity called entropy: consider a stationary, ergodic process (as an example of such an object, think of a ticker-tape printing out 0s and 1s randomly according to some probability law). Order the words[1] of length n coming from this process in decreasing order (in terms of their probabilities). After fixing $\lambda \in (0, 1)$, select words of length n in the above order, one at a time starting with the most likely word, until the probabilities of the selected words sum to at least λ. Defining $N_n(\lambda)$ to be the number of words it takes to do this, it turns out that

$$\lim_{n \to \infty} \frac{1}{n} \log N_n(\lambda) = h \tag{1.1}$$

where h is a number, independent of λ, called the **entropy** of the stochastic process. The quantity h measures the amount of "randomness" or "chaos" in the process, and is an important invariant of stationary processes which has applications in data compression and ergodic theory. Essentially, the fact in Equation (1.1) is a corollary of what is known in information theory as the Shannon-McMillan-Breiman

[1] In the ticker-tape example, a "word" of length 6 would be something like 001011 or 110111.

Theorem, or the "asymptotic equipartition property" (see [Sh, M, B, AC] for more on this).

Remark: The base of the logarithm is irrelevant in the definition of entropy, as choosing two different bases yields the same answer up to a constant multiple. We choose base e, but the only time in our paper that this matters is during the proof of Theorem 3.1.

1.6. **History and summary of our main results.** Several mathematicians and computer scientists have done work on counting numbers of various LEGO configurations made from rectangular bricks [AE, DE1, DE2, JKC, KKC, Li]. As mentioned earlier, Durhuus and Eilers showed in [DE2] that the number $T_{b\times w}(n)$ of buildings that can be made from n standard rectangular $b \times w$ LEGO bricks grows exponentially in n, and described upper and lower bounds on the entropy of 2×4 bricks.

In this paper, we investigate the entropy of LEGO jumper plates, using some methods borrowed from [DE2] and other methods involving the combinatorics of objects we call "labelled binary graphs". In the next two sections, we show that the limit defining the entropy exists and is at least $\log 6.44947$. The techniques in these sections are borrowed heavily from Durhuus and Eilers, who studied the entropy of (non-jumper) rectangular LEGO bricks in [DE2]. In Section 4 we prove that the entropy is at most $\log(6+\sqrt{2})$, using a new method of associating a "labelled binary tree" to each building and counting the number of such labelled trees. The method of associating a graph to a LEGO construction was used in [DE1], but the idea of labelling the graphs (and the associated combinatorics) is, as far as we know, new.

Section 5 contains an outline of how our methods might be further improved, and the last section outlines how our methods can be applied to a different type of LEGO element called a "roof tile".

2. Existence of entropy

Durhuus and Eilers established the existence of the entropy for configurations of rectangular $b \times w$ bricks in [DE2]; we mimic their argument to explain why $h_{\mathcal{J}}$ exists.

Theorem 2.1. *Let $T_{\mathcal{J}}(n)$ be the number of buildings made from n 1×2 jumper plates. Then*

$$h_{\mathcal{J}} = \lim_{n\to\infty} \frac{1}{n} \log T_{\mathcal{J}}(n)$$

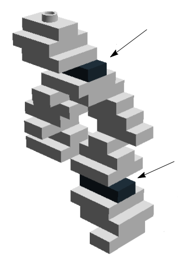
exists in $[0, \infty]$.

Proof. Denote by $B_{\mathcal{J}}(n)$ the set of buildings which can be made from n jumper plates. Then, let $A_{\mathcal{J}}$ be the subset of $B_{\mathcal{J}}$ consisting of buildings whose bottom-most layer contains exactly one jumper plate; let $a_n = \#(A_{\mathcal{J}}(n))$. Observe that

$$T_{\mathcal{J}}(n-1) \leq a_n \leq T_{\mathcal{J}}(n). \tag{2.1}$$

To see the left-hand inequality, notice that by removing the bottom plate from each member of $A_{\mathcal{J}}(n)$, we obtain a member of $B_{\mathcal{J}}(n-1)$, and every member of $B_{\mathcal{J}}(n-1)$ can be obtained in this fashion. The right-hand inequality follows from the fact that $A_{\mathcal{J}}(n) \subseteq B_{\mathcal{J}}(n)$.

Figure 8. A LEGO building with two bottlenecks, located at the black jumper plates indicated by the arrows. Each of the black jumper plates is the only plate in its level of the building.



From (2.1), we see that

$$h_{\mathcal{J}} = \lim_{n\to\infty} \frac{1}{n} \log T_{\mathcal{J}}(n) = \lim_{n\to\infty} \frac{1}{n} \log a_n.$$

Next, notice that $a_{n+m} \geq a_n a_m$. To see why, observe that by attaching the root of any element of $A_{\mathcal{J}}(n)$ to the underside of the plate on the bottom level of any building in $A_{\mathcal{J}}(m)$ produces a building in $A_{\mathcal{J}}(m+n)$. This procedure yields an injection $A_{\mathcal{J}}(n) \times A_{\mathcal{J}}(m) \hookrightarrow A_{\mathcal{J}}(m+n)$, giving the desired inequality. Therefore, for all m and n, $\log a_{m+n} \geq \log a_m + \log a_n$, so by Fekete's lemma $\left\{\frac{1}{n}\log a_n\right\}$ converges as $n \to \infty$ to $\sup\left\{\frac{1}{n}\log a_n\right\} \in [0,\infty]$. □

3. A lower bound on the entropy

First, as there are 6 choices for how an only child can be attached to its parent, the number of buildings of height n that can be made from n jumper plates is 6^{n-1}, thus producing the trivial lower bound

$$h_{\mathcal{J}} \geq \lim_{n\to\infty} \frac{1}{n} \log 6^{n-1} = \log 6.$$

We tighten this bound by applying a technique developed in [DE2] which counts the number of buildings with a fixed number of "bottlenecks".

Theorem 3.1. *Let $h_{\mathcal{J}}$ be the entropy of a 1×2 LEGO jumper plate. Then*

$$h_{\mathcal{J}} \geq \log 6.44947.$$

Proof. Recall $A_{\mathcal{J}}(n+1)$ is the set of LEGO buildings made from $n+1$ jumper plates such that the top and bottom layer of the building each consist of a single plate. We say that a building in $A_{\mathcal{J}}(n+1)$ has a **bottleneck** if the building has a layer, other than the top and/or bottom layer, which has only a single brick in it (see Figure 8).

For $n \geq 0$, let c_n denote the number of buildings in $A_{\mathcal{J}}(n+1)$ that have no bottlenecks. In [DE2], the authors show using generating functions that for any n,

$$\sum_{j=1}^{n} c_j \left(e^{h_{\mathcal{J}}}\right)^{-j} \leq 1, \tag{3.1}$$

and their argument carries over into our context. By explicitly computing values of c_j for some j, we obtain a lower bound for $h_{\mathcal{J}}$. First, it is clear that $c_1 = 6$, since any building made from two plates has no bottleneck. Note also that $c_2 = 0$, because any building with three plates arranged in three layers must have a bottleneck in the middle layer.

To determine c_3, note that any building in $A_{\mathcal{J}}(4)$ without bottlenecks must have one plate in its bottom layer, and two in the middle layer which are both children of the root. There is therefore only one way to hook the top three plates together, so c_3 is equal to the number of ways to attach a single jumper plate to the bottom of one of two parallel jumper plates. There are six ways to attach this last plate to its parent, and two choices of parent, so $c_4 = 6(2) = 12$.

Next, $c_4 = 0$, because any building made from five jumper plates where the top and bottom layers consist of a single plate must have a bottleneck in it.

For c_5, observe that any building in $A_{\mathcal{J}}(6)$ with no bottlenecks must have one plate in the top layer (call this layer 0), one plate in the bottom layer (layer 3), and two plates in each of the two intermediate layers (layers 1 and 2). Therefore there is one way to configure the three jumper plates in layers 0 and 1. Once those three are attached, two more jumper plates need to be attached underneath layer 1 to form layer 2. There are 9 ways to do this so that the plates in layer 2 are parallel to the plates in layer 1, and 4 ways to do this so that the plates in layer 2 are perpendicular to the plates in layer 1. Once layer 2 is made, the last plate (which comprises layer 3) needs to be attached to the bottom of one of the two plates in layer 2 to finish the building; there are 12 ways to do this. Altogether, $c_5 = (4 + 9)12 = 156$.

At this point we know from (3.1) that

$$6\left(e^{h_{\mathcal{J}}}\right)^{-1} + 12\left(e^{h_{\mathcal{J}}}\right)^{-3} + 156\left(e^{h_{\mathcal{J}}}\right)^{-5} \leq 1,$$

from which it follows that $h_{\mathcal{J}} \geq \log 6.3877$.

S. Eilers [E] relayed to us computer-generated computations of $c_7 = 2652$, $c_8 = 144$, $c_9 = 59100$, $c_{10} = 18192$ and $c_{11} = 1615740$; applying these values, the lower bound improves to $h_{\mathcal{J}} \geq \log 6.44947$. □

Notice that this lower bound is greater than the value of $h_{\mathcal{J}}$ suggested by the least-squares computation in Section 1 (which was $\log 6.39368$).

4. An upper bound on the entropy

In this section we look for an upper bound on $h_{\mathcal{J}}$. Remember from the proof of Theorem 2.1 that $B_{\mathcal{J}}(n)$ denotes the set of LEGO buildings made from n jumper plates.

4.1. **A crude upper bound.** We obtain $h_{\mathcal{J}} \leq \log 8$ by applying a method described in [DE2] which associates to each LEGO building a string of characters taken from a finite alphabet. More specifically, let $b \in B_{\mathcal{J}}(n)$ be a building. Number the plates in b from 1 to $n-1$ as follows: call the root of the building "plate 1", then number the child(ren) of the root "plate 2" (and "plate 3", if the root has two children), then continue inductively, numbering the children of plate 2, then any children of plate 3, etc. Any time that a plate has two children, choose a standard way to order the children (for example, choose a compass direction to represent north, and whenever a plate has two children, give the smaller number to the plate that is either further south or further west, depending on its orientation).

Next, number the different ways to connect two jumper plates 1 to 6 as shown in Figure 3, and define $\mathcal{A} = \{0, \bowtie, 1, 2, 3, 4, 5, 6\}$.

Then define $f : B_{\mathcal{J}}(n) \hookrightarrow \mathcal{A}^{n-1}$ as follows: $f(b) = (x_1, ..., x_{n-1})$ if, for every $j \in \{1, ..., n-1\}$,

$$x_j = \begin{cases} 0 & \text{if plate } j \text{ of the building has no children} \\ \bowtie & \text{if plate } j \text{ of the building has two children} \\ z \in \{1, ..., 6\} & \text{if plate } j \text{ of the building has exactly one child, which is} \\ & \quad \text{attached to its parent via connection } z \text{ as shown in} \\ & \quad \text{Figure 3.} \end{cases}$$

In the last case above, to distinguish between connections like those numbered 1 and 3 in Figure 3, one can decree that if the child is attached to the southernmost or westernmost slot of the parent, then the connection is type 1; otherwise it is type 3.

Essentially, the symbol in the j^{th} position of $f(b)$ tells you how to attach children to the j^{th} plate in the building b. As such, a string of $n-1$ symbols provides directions to construct at most one building, so f is $1-1$. Thus

$$T_{\mathcal{J}}(n) = \#(B_{\mathcal{J}}(n)) \leq \#(\mathcal{A}^{n-1}) = 8^{n-1}$$

and it follows that

$$h_{\mathcal{J}} \leq \lim_{n\to\infty} \frac{1}{n} \log 8^{n-1} = \log 8.$$

4.2. **Bounding the entropy by counting labelled binary trees.** In this section, we improve the upper bound to $h_{\mathcal{J}} \leq \log\left(6 + \sqrt{2}\right)$ by a new method which associates, to each LEGO building, a "labelled binary tree", counting the number of such labelled trees with specific properties, and counting the maximum number of buildings that can be associated to each such labelled tree.

First, by a **binary tree** $\mathcal{T}$, we mean a full binary tree where the left and right children at each node are distinguished. More formally, we decree a binary tree to be a rooted tree which is also an ordered tree, where every node has either 0 or 2 children. Given such a binary tree, a **branching** of the tree is a node which has 2 children. We denote the set of nodes of binary tree $\mathcal{T}$ by $V(\mathcal{T})$.

Next, a **labelled binary tree** is a pair $(\mathcal{T}, f)$ where $\mathcal{T}$ is a binary tree and f is a function which assigns to each node in $\mathcal{T}$ a positive integer. For each $n \in \{1, 2, 3, ...\}$, let L_n be the set of labelled binary trees $(\mathcal{T}, f)$ such that

$$\sum_{v \in V(\mathcal{T})} f(v) = n;$$

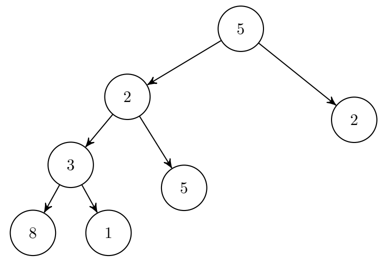
denote the cardinality of L_n by $Q(n)$. For each $n \in \{1, 2, 3, ...\}$ and $k \in \{0, 1, 2, ...\}$, define $L_{n,k}$ to be the set of labelled binary trees in L_n which have exactly k branchings. See Figure 9 for an example.

The first key observation related to our counting of LEGO structures is this:

Theorem 4.1. *Let $T_{\mathcal{J}}(n)$ be the number of LEGO buildings that can be made from n 1×2 jumper plates. Then*

$$T_{\mathcal{J}}(n) \leq \sum_{k=0}^{\left\lfloor \frac{n-1}{2} \right\rfloor} 6^{n-1-2k} \#(L_{n,k}).$$

FIGURE 9. A labelled binary tree belonging to the set $L_{26,3}$. (The values of the labelling function f are written inside each circle. 26 is the sum of the labels on the nodes; 3 is the number of branchings.)



Proof. We begin by associating to each building $b \in B_{\mathcal{J}}(n)$ a labelled binary graph $\Theta(b) \in L_{n,k}$. The idea of how this association works is shown in Figure 10; the concept is that the nodes of tree $\Theta(b)$ indicate the parents in the original building which have 2 children, and the labels on the nodes (i.e. the values of f) indicate how many generations one needs to pass through before seeing the next plate with two children. Now for the details:

First, given $b \in B_{\mathcal{J}}(n)$, associate a graph to b by thinking of the individual jumper plates comprising b as nodes and saying that the nodes are related if the corresponding plates are attached, similar to what was done in [DE1]. This produces a tree $\theta(b)$ whose root corresponds to the root of the building, where each node in $\theta(b)$ has at most 2 children.

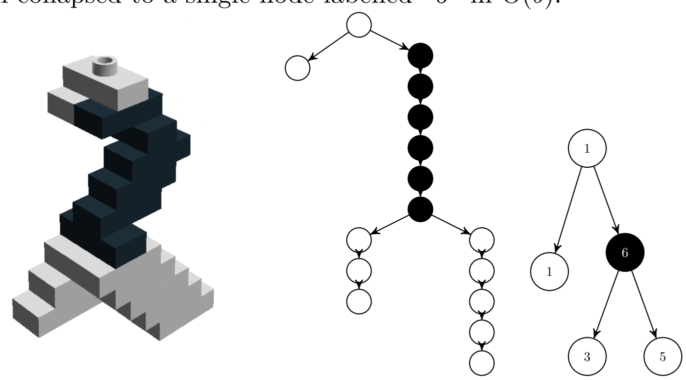
To order the tree $\theta(b)$, we need to consider the situation where a plate in the building has two children. To do this, choose a compass direction to represent north. If a plate has two children, either one child is south of the other, or one child is west of the other. In the first case, decree the left branch in $\theta(b)$ to correspond to the southernmost child, and in the second case, decree the left branch in $\theta(b)$ to correspond to the westernmost child.

To obtain the labelled binary tree $\Theta(b)$, we next define an equivalence relation on the nodes of $\theta(b)$. Given nodes v and w in $\theta(b)$, say $v \preceq w$ if there is a chain of nodes $v = v_0, v_1, v_2, ..., v_n = w$ such that for each $j \in \{1, 2, 3, ..., n\}$, v_j is the only child of v_{j-1}. (Notice that for all nodes v, $v \preceq v$ by setting $n = 0$.) Then declare nodes v and w to be equivalent if $v \preceq w$ or $w \preceq v$.

Denoting the equivalence class of a node v under this relation by $[v]$, we obtain a labelled binary tree $\Theta(b)$ whose vertices are the equivalence classes $[v]$, whose edge relations are defined by saying $[w]$ is the child of $[v]$ if and only if some member of $[w]$ is the child of some member of $[v]$ in tree $\theta(b)$, and whose labelling function f is defined by $f([a]) = \#([a])$. This completes the formal definition of Θ.

For the second part of the proof, we count the maximum number of preimages a labelled binary tree has under Θ. If $\Theta(b) \in L_{n,k}$, then b must be a building made of n jumper plates, of which k have exactly two children; these plates are at locations specified by the labels of $\Theta(b)$. To describe such a building, one therefore needs only to specify how to attach the jumper plates which are only children. There

Figure 10. This figure shows an example of how we associate a labelled binary tree to a LEGO building made from jumper plates. On the left, we have a LEGO building b made from 16 jumper plates. In the center, its binary tree $\theta(b)$ is shown. This binary tree essentially "forgets" the orientation of the plates and simply records how the individual pieces are attached. In particular, the nodes colored black in this tree come from the plates colored black in the building. On the right, the labelled binary tree $\Theta(b) \in L_{16,2}$ is pictured; note that the six individual black nodes in $\theta(b)$ have been collapsed to a single node labelled "6" in $\Theta(b)$.



$b \in B_{\mathcal{J}}(16)$ $\theta(b)$ $\Theta(b) \in L_{16,2}$

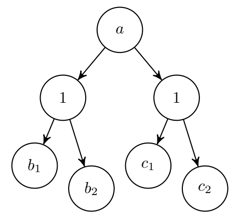
are $n-1-2k$ such plates which need to be attached to make such a building, and for each such plate there are ≤ 6 ways to attach the plate to its parent. Thus there are at most 6^{n-1-2k} buildings b for which $\Theta(b)$ is a fixed labelled tree in $L_{n,k}$. Summing this count from the minimum number of branchings (zero) to the maximum number of branchings in a building made from n jumper plates ($\lfloor \frac{n-1}{2} \rfloor$) gives the inequality of the theorem. $\square$

In light of Theorem 4.1, one way we could find an upper bound on $T_{\mathcal{J}}(n)$ would be to study the growth rate of the sequence $\#(L_{n,k})$. But in fact, we can do better by observing that the function Θ defined in Theorem 4.1 is very far from surjective. As an example, suppose that some jumper plate in a building has 2 children. Because these two children must be parallel and share a common boundary of length 2, it is impossible for either of those two children to themselves have two children unless their sibling is childless. So in a building made from jumper plates, no plate can have more than 2 grandchildren, meaning, for example, that a labelled graph such as the one shown in Figure 11 cannot be $\Theta(b)$ for any $b \in B_{\mathcal{J}}(n)$.

Defining $L^*_{n,k}$ to be the set of labelled binary graphs in $L_{n,k}$ which are actually obtained as $\Theta(b)$ for at least one building $b \in B_{\mathcal{J}}(n)$, and denoting the cardinality of the set $L^*_{n,k}$ as $Q(n,k)$, it follows from the reasoning in the last paragraph of the proof of Theorem 4.1 that

$$T_{\mathcal{J}}(n) \leq \sum_{k=0}^{\lfloor \frac{n-1}{2} \rfloor} 6^{n-1-2k} Q(n,k). \tag{4.1}$$

FIGURE 11. No labelled graph of this type (where a, b_1, b_2, c_1 and c_2 are positive integers) can be obtained as $\Theta(b)$ for any building $b \in B_{\mathcal{J}}(n)$, because the plate at the top-most branching would have 4 grandchildren.



Instead of studying the growth rate of $\#(L_{n,k})$, we instead will find an effective upper bound on the size of $Q(n,k)$. The next three lemmas work toward this goal. In Lemma 4.2, we lay out some preliminary properties of $Q(n,k)$. Lemma 4.3 establishes a recursively defined upper bound for $Q(n,k)$, and Lemma 4.4 uses the preceding two lemmas to establish a closed formula for a nice upper bound on $Q(n,k)$.

Lemma 4.2 (Properties of $Q(n,k)$)**.** *Let $Q(n,k)$ be defined as above. Then:*

(1) *If $n < 2k+1$, then $Q(n,k) = 0$.*
(2) *For any $n \in \{1,2,3,...\}$, $Q(n,0) = 1$.*
(3) *For any $k \in \{1,2,...\}$, $Q(2k+1,k) = 2^{k-1}$.*

Proof. For (1), observe that a (full) binary tree $\mathcal{T}$ with k branchings must have exactly $2k+1$ nodes. Thus, for any function $f : V(\mathcal{T}) \to \{1,2,3,...\}$, $\sum_{v \in V(\mathcal{T})} f(v) \geq 2k+1$, so no pair $(\mathcal{T}, f)$ can exist in $L_{n,k}$ if $n < 2k+1$.

For (2), we note that a tree with zero branchings consists of a single node. The only element of $L_{n,0}$ is therefore this single node, together with the function assigning n to that node.

Last, to show (3), notice that a labelled binary tree belongs to $L_{2k+1,k}$ if and only if the tree has k branchings and $f(v) = 1$ for every node in $V(\mathcal{T})$. In order for such a tree to come from a building made from jumper plates, there must be only one branching at each level of the tree (otherwise, there would be a plate in the building with four grandchildren, which is impossible). This means that at each level other than the root, there are two choices for which child in the tree has a branching (the left or the right). Since there are $k-1$ branchings other than the one at the root, we obtain $Q(2k+1,k) = 2^{k-1}$ as wanted. □

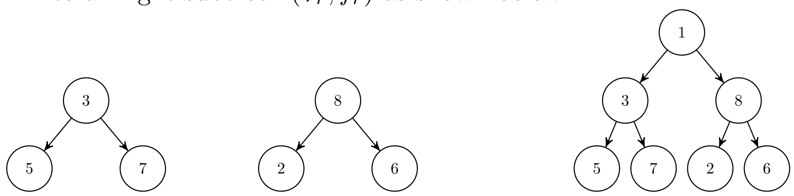
Lemma 4.3 (Recursive upper bound for $Q(n,k)$)**.** *For any $n \in \{1,2,3,...\}$ and any $k \in \{0,1,2,...\}$,*

$$Q(n,k) \leq Q(n-1,k) + \sum_{j=0}^{n-1} \sum_{s=0}^{k-1} Q(j,s)Q(n-j-1,k-s-1).$$

Proof. Let $L_{n,k}^{(1)}$ be the set of labelled binary trees in $L_{n,k}^{*}$ such that f assigns 1 to the root vertex.

First, we count the complement of $L_{n,k}^{(1)}$. To do this, observe that any such tree can be associated to a tree in $L_{n-1,k}^{*}$ by subtracting 1 from the label on the root.

Figure 12. An example of how we join labelled trees. We count the number of labelled trees in $L^{(1)}_{n,k}$ by observing that every labelled tree in that set comes from joining a "left subtree" $(\mathcal{T}_l, f_l)$ to a "right subtree" $(\mathcal{T}_r, f_r)$ as shown below.



$(\mathcal{T}_l, f_l)$ (element of $L^*_{15,1}$) $(\mathcal{T}_r, f_r)$ (element of $L^*_{16,1}$) $J = J((\mathcal{T}_l, f_l), (\mathcal{T}_r, f_r))$ (element of $L^{(1)}_{15+16,1+1+1} = L^{(1)}_{31,3}$)

More precisely, for any $(\mathcal{T}, f) \in L^*_{n,k} - L^{(1)}_{n,k}$, define $g : V(\mathcal{T}) \to \{1, 2, 3, ...\}$ by

$$g(v) = \begin{cases} f(v) & \text{if } v \text{ is not the root of } \mathcal{T} \\ f(v) - 1 & \text{if } v \text{ is the root of } \mathcal{T} \end{cases}$$

The mapping $(\mathcal{T}, f) \mapsto (\mathcal{T}, g)$ therefore gives a bijection between $L^*_{n,k} - L^{(1)}_{n,k}$ and $L^*_{n-1,k}$, so $\#\left(L^*_{n,k} - L^{(1)}_{n,k}\right) = Q(n-1, k)$.

Next, we count $L^{(1)}_{n,k}$. Let $j \in \{0, 1, ..., n-1\}$ and let $s \in \{0, 1, ..., k-1\}$. Given any two labelled trees $(\mathcal{T}_l, f_l) \in L^*_{j,s}$ and $(\mathcal{T}_r, f_r) \in L^*_{n-j-1,k-s-1}$, we can "join" those trees together to create a tree in $L^{(1)}_{n,k}$ as shown in Figure 12. More precisely, we build a labelled tree $J = J\left((\mathcal{T}_l, f_l), (\mathcal{T}_r, f_r)\right)$ as follows: the tree $\mathcal{T}$ of J is formed by taking trees $\mathcal{T}_l$ and $\mathcal{T}_r$, adding one more node to serve as the root of the new tree, and decreeing the roots of $\mathcal{T}_l$ and $\mathcal{T}_r$ to be, respectively, the left and right children of the new root. The function $f : V(\mathcal{T}) \to \{1, 2, 3, ...\}$ assigns 1 to the root of $\mathcal{T}$, and agrees with f_l and f_r on $\mathcal{T}_l$ and $\mathcal{T}_r$. Notice that

$$\begin{aligned} \sum_{v \in V(\mathcal{T})} f(v) &= f(root(\mathcal{T})) + \sum_{v \in V(\mathcal{T}_l)} f_l(v) + \sum_{v \in V(\mathcal{T}_r)} f_r(v) \\ &= 1 + j + (n - j - 1) \\ &= n \end{aligned}$$

and the number of branchings in J is 1 (from the new root) plus s (the number of branchings in $\mathcal{T}_l$) plus $k - s - 1$ (the number of branchings in $\mathcal{T}_r$). Therefore $J \in L_{n,k}$; of course, J might not be in the image of Θ, but every building in $L^{(1)}_{n,k}$ can be obtained in this way, so we know

$$\#\left(L^{(1)}_{n,k}\right) \leq \sum_{j=0}^{n-1} \sum_{s=0}^{k-1} Q(j, s) Q(n - j - 1, k - s - 1).$$

Adding together the counts of $L^{(1)}_{n,k}$ and $L^*_{n,k} - L^{(1)}_{n,k}$ gives the inequality in the statement of the lemma. □

The "initial values" of $Q(n, k)$ given in Lemma 4.2 and the recursive formula of Lemma 4.3 can be combined to obtain the following upper bound on $Q(n, k)$:

Lemma 4.4 (Upper bound on $Q(n,k)$)**.** *Let $Q(n,k)$ be defined as above. Then*

$$Q(n,k) \le \binom{n-1}{2k} 2^{k-1}.$$

Proof. For $n \ge 1$ and $k \ge 0$, define numbers $R(n,k)$ by using the information from Lemma 4.2 and pretending that the inequality given in Lemma 4.3 is actually an equality: more formally, set

$$R(n,k) = 0 \text{ for any } n < 2k+1; \tag{4.2}$$

$$R(n,0) = 1 \text{ for any } n \ge 0; \tag{4.3}$$

$$R(2k+1,k) = 2^{k-1} \text{ for any } k \ge 1; \tag{4.4}$$

and recursively define, for any $n > 2k+1$,

$$R(n,k) = R(n-1,k) + \sum_{j=0}^{n-1} \sum_{s=0}^{k-1} R(j,s) R(n-j-1, k-s-1). \tag{4.5}$$

In light of Lemma 4.3, $Q(n,k) \le R(n,k)$ for all n and k. We will prove the lemma by showing $R(n,k) = \binom{n-1}{2k} 2^{k-1}$.

The key to this lemma is to see that for each $n \in \{1,2,3,...\}$ and each $k \in \{0,1,2,...\}$, $R(n,k)$ is a polynomial of degree $2k$ in the variable n. To prove this claim, we use induction on k. The base case $k = 0$ is obvious since $R(n,0) = 1$ for all $n \ge 1$.

For the induction step, fix $k > 0$ and assume that for all $j < k$, $R(n,j)$ is a degree $2j$ polynomial in the variable n. Now define $R(0,k) = 0$ and for $n \in \{1,2,3,...\}$, set $D(n,k) = R(n,k) - R(n-1,k)$. By the formula (4.5) above, we see that

$$D(n,k) = \sum_{j=0}^{n-1} \sum_{s=0}^{k-1} R(j,s) R(n-j-1, k-s-1).$$

By the induction hypothesis, $R(j,s)$ is a polynomial of degree $2s$ in the variable j and $R(n-j-1, k-s-1)$ is a polynomial of degree $2(k-s-1)$ in the variable $n-j-1$. Therefore, for each j and s, the expression

$$R(j,s) R(n-j-1, k-s-1)$$

is a polynomial in two variables n and j (degree $2(k-1)$ in the variable j and $2(k-s-1)$ in the variable n). When these polynomials are summed from $s = 0$ to $k-1$, we obtain a polynomial which is degree $2(k-1) = 2k-2$ in the variable j and degree $2(k-1)$ in the variable n. Therefore $D(n,k)$, being the sum from $j = 0$ to $n-1$ of such polynomials, is a polynomial of degree $2k-1$ in the variable n (the highest degree coming from the polynomials in variable j being added together). Finally,

$$R(n,k) = \sum_{j=0}^{n} D(j,k)$$

is the sum of $n+1$ polynomials of degree $2k-1$ in the variable n, which is a polynomial of degree $2k$. This establishes the claim.

At this point, we know that $R(n,k)$ is a polynomial of degree $2k$ which has roots when $n = 1,2,3,4,...,2k$. Therefore, for some constant C depending on k,

$$R(n,k) = C(n-1)(n-2)\cdots(n-2k)$$

and in particular, $R(2k+1,k) = C(2k)!$. But we know $R(2k+1,k) = 2^{k-1}$ by (4.4), and it follows that $C = \frac{2^{k-1}}{(2k)!}$. Therefore

$$\begin{aligned} R(n,k) &= \frac{2^{k-1}}{(2k)!}(n-1)(n-2)\cdots(n-2k) \\ &= \frac{2^{k-1}}{(2k)!}\frac{(n-1)!}{(n-2k-1)!} \\ &= 2^{k-1}\binom{n-1}{2k} \end{aligned}$$

as needed. □

To summarize, at this point we know by combining Theorem 4.1 and Lemma 4.4 that the number $T_{\mathcal{J}}(n)$ of buildings made from n LEGO jumper plates satisfies

$$\begin{aligned} T_{\mathcal{J}}(n) &\leq \sum_{k=0}^{\lfloor \frac{n-1}{2} \rfloor} Q(n,k)6^{n-1-2k} \\ &\leq \sum_{k=0}^{\lfloor \frac{n-1}{2} \rfloor} \binom{n-1}{2k} 2^{k-1} \cdot 6^{n-1} \left(\frac{1}{36}\right)^k. \end{aligned}$$

Using the convention that $\binom{n}{k} = 0$ when $n < k$, this inequality can be rewritten as

$$T_{\mathcal{J}}(n) \leq \frac{1}{2} \cdot 6^{n-1} \sum_{k=0}^{\infty} \binom{n-1}{2k} \left(\frac{1}{18}\right)^k. \tag{4.6}$$

We have obtained a series on the right-hand side of (4.6) which, fortunately, can be summed using the binomial theorem:

Lemma 4.5. *For any $r \in (0,1)$,*

$$\sum_{k=0}^{\infty} \binom{n-1}{2k} r^k = \frac{(1+\sqrt{r})^{n-1} + (1-\sqrt{r})^{n-1}}{2}.$$

Proof. Let $r \in (0,1)$. From the binomial theorem,

$$\begin{aligned} (1+\sqrt{r})^{n-1} &= \sum_{k=0}^{\infty} \binom{n-1}{k} \left(\sqrt{r}\right)^k (1)^{n-1-k} \\ &= \binom{n-1}{0} + \binom{n-1}{1} r^{1/2} + \binom{n-1}{2} r + \binom{n-1}{3} r^{3/2} + \ldots \end{aligned}$$

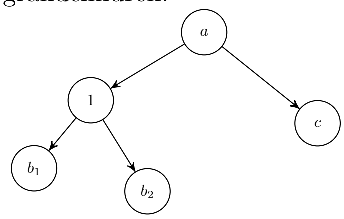
and also

$$\begin{aligned} (1-\sqrt{r})^{n-1} &= \sum_{k=0}^{\infty} \binom{n-1}{k} (-\sqrt{r})^k (1)^{n-1-k} \\ &= \binom{n-1}{0} - \binom{n-1}{1} r^{1/2} + \binom{n-1}{2} r - \binom{n-1}{3} r^{3/2} + \ldots \end{aligned}$$

When the two preceding series are added together, the non-integer powers of r cancel; dividing the sum by 2 gives the formula in the claim. □

Finally, we are able to put all the work of this section together and deduce the upper bound on $h_{\mathcal{J}}$:

FIGURE 13. If $c > 1$, then no matter the values of positive integers a, b_1 and b_2, this labelled graph cannot be $\Theta(b)$ for any $b \in B_{\mathcal{J}}(n)$ for any n, because the plate corresponding to the top-most branching would have 3 grandchildren.



Theorem 4.6. *The entropy $h_{\mathcal{J}}$ of a 1×2 jumper plate satisfies $h_{\mathcal{J}} \leq \log\left(6 + \sqrt{2}\right)$.*

Proof. Applying the formula in Lemma 4.5 to Equation (4.6), we get

$$\begin{aligned} T_{\mathcal{J}}(n) &\leq 6^{n-1}\left(\frac{1}{4}\right)\left[\left(1+\sqrt{\frac{1}{18}}\right)^{n-1} + \left(1-\sqrt{\frac{1}{18}}\right)^{n-1}\right] \\ &= \frac{1}{4}\left[\left(6+\sqrt{2}\right)^{n-1} + \left(6-\sqrt{2}\right)^{n-1}\right] \end{aligned}$$

and therefore

$$\begin{aligned} h_{\mathcal{J}} &= \lim_{n\to\infty} \frac{1}{n}\log T_{\mathcal{J}}(n) \\ &\leq \lim_{n\to\infty} \frac{1}{n}\log\frac{1}{4}\left[\left(6+\sqrt{2}\right)^{n-1} + \left(6-\sqrt{2}\right)^{n-1}\right]. \end{aligned}$$

The dominant exponential term inside the brackets comes from the $\left(6+\sqrt{2}\right)^{n-1}$ term, so we obtain $h_{\mathcal{J}} \leq \log\left(6+\sqrt{2}\right)$ as desired. (Alternatively, this limit can be evaluated rigorously using logarithm rules and L'Hôpital's Rule.) □

5. Further improvements to the upper bound

Recall from the discussion in Section 4 that many labelled binary graphs in $L_{n,k}$ cannot be obtained as $\Theta(b)$ for any building b made from n jumper plates. We obtained the upper bound $h_{\mathcal{J}} \leq \log\left(6+\sqrt{2}\right)$ by throwing out some labelled binary graphs for which some nodes have four grandchildren. However, this technique does not come close to discarding <u>all</u> the labelled binary graphs which are not in the range of Θ. In particular, any labelled binary graph containing a subgraph like either of those in Figures 11 or 13 cannot be $\Theta(b)$ for any building b.

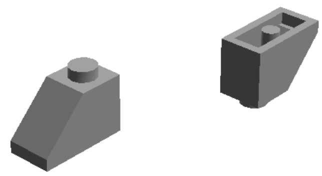
We propose a program to further improve our upper bound as follows: as before, for each n and k, let

$$L^*_{n,k} = \mathit{Range}(\Theta) \cap L_{n,k}$$

and let $Q^*(n,k)$ be any sequence satisfying $Q^*(n,k) \geq \#(L^*_{n,k})$. By the same argument as the one given in Section 4, we have

$$h_{\mathcal{J}} \leq \sum_{k=0}^{\left\lfloor \frac{n-1}{2} \right\rfloor} Q^*(n,k) 6^{n-1-2k}.$$

FIGURE 14. The top and bottom of a 2×1 LEGO roof tile.



so any $Q^*(n,k)$ which grows more slowly than ours (in this language, what we used for "$Q^*(n,k)$" in Section 4 was $R(n,k) = \binom{n-1}{2k}2^k$) would improve the upper bound. One way to do this is would be to count the number of labelled binary graphs coming from binary graphs in which no node has more than two grandchildren. By itself, however, this improvement would not give the exact value of $Q^*(n,k)$, as there are other restrictions on the kinds of labelled binary graphs which lie in the range of Θ.

6. Remarks on the entropy of LEGO roof tiles

S. Eilers asked whether our methods can be adapted to study 2×1 LEGO roof tiles (such a piece, which we denote by class $\mathcal{R}$, slopes inward at a 45° angle from a base which measures 1×2 to a top that measures 1×1; see Figure 14). Unlike jumper plates, the bottom of a roof tile allows only two slots for the top stud of a child to be inserted into its parent.

First, the argument given in Section 2 for jumper plates carries over directly to show that $h_{\mathcal{R}} = \lim_{n\to\infty} \frac{1}{n} \log T_{\mathcal{R}}(n)$ exists. Second, it's easy to see that for 2×1 roof tiles, $T_{\mathcal{R}}(2) = 8$, because to connect two such pieces together, one needs to choose a slot to insert the child (2 options) and independently choose a direction for the slope of the child to point (4 options). Thus a crude lower bound on the entropy is $h_{\mathcal{R}} \geq \log 8$, and this bound could be improved by methods akin to what we executed for jumper plates in Section 3 (for this class of brick, $c_1 = 8$ and $c_3 = 144$, producing a lower bound of $h_{\mathcal{R}} \geq \log 9.57174$).

As for an upper bound on $h_{\mathcal{R}}$, by associating a labelled graph $\Theta(b)$ to any building b made from n 2×1 roof plates in the same way we did for jumper plates in Section 4, one obtains

$$T_{\mathcal{R}}(n) \leq \sum_{k=0}^{\lfloor \frac{n-1}{2} \rfloor} 8^{n-1-2k} 9^k \#(L_{n,k}).$$

The extra 9^k term in this expression comes from the fact that there are 9 distinct ways to attach two roof tiles underneath a particular roof tile, and a building associated to a graph in $L_{n,k}$ has exactly k of these branchings.

Unfortunately, at this point the problem becomes more difficult, because unlike jumper plates, a roof plate may have 4 grandchildren. Our methods do provide an upper bound of $h_{\mathcal{R}} \leq \log 14$, but with far more complicated analysis using special mathematical functions which we briefly outline in the next paragraph, leaving the details for the interested reader to work out.

Using methods similar to what we did in Lemmas 4.2 to 4.4, one can show that that

$$\#(L_{n,k}) = \binom{n-1}{2k}\binom{2k}{k}\frac{1}{k+1}$$

and therefore

$$\begin{aligned} T_{\mathcal{R}}(n) &\le 8^{n-1}\sum_{k=0}^{\lfloor \frac{n-1}{2}\rfloor}\binom{n-1}{2k}\binom{2k}{k}\frac{1}{k+1}\left(\frac{9}{64}\right)^k \\ &= 8^{n-1}\sum_{k=0}^{\infty}\frac{(n-1)!}{(n-2k-1)!(k+1)!}\,\frac{(9/64)^k}{k!} \\ &= 8^{n-1}\,{}_2F_1\left(\frac{1}{2}-\frac{n}{2},1-\frac{n}{2},2;\frac{9}{16}\right) \end{aligned}$$

where ${}_2F_1$ is the Gauss hypergeometric function [OD, Sl]. Using a symmetry of the hypergeometric function called the Pfaff transformation (see, for example, [KO]), and the definition of the Jacobi polynomial $P_m^{(a,b)}$ (see Chapter 4 of [Sz]), this upper bound can be rewritten as

$$8^{n-1}\left(\frac{7}{16}\right)^m\frac{1}{m+1}P_m^{(1,\frac{-1}{2})}\left(\frac{25}{7}\right)$$

where $m = \frac{1}{2}(n-1)$. For large m, this Jacobi polynomial can subsequently be approximated using Darboux's formula (see formula (1.2) of [BG]) as

$$P_m^{(1,\frac{-1}{2})}\left(\frac{25}{7}\right) = \frac{K}{\sqrt{m}}\,7^{m+1}\,(1+o(1))$$

for a suitable constant K. Putting all this together,

$$T_{\mathcal{R}}(n) \le 8^{n-1}\left(\frac{7}{16}\right)^{\frac{n-1}{2}}7^{\frac{n+1}{2}}\frac{K}{(m+1)\sqrt{m}}\,(1+o(1))$$

and by taking the logarithm of this expression, dividing by n and letting $n\to\infty$, one obtains the upper bound

$$h_{\mathcal{R}} \le \log 8 + \log\frac{\sqrt{7}}{4} + \log\sqrt{7} = \log 14.$$

It would be interesting to study other LEGO bricks for which an upper bound on their entropy can be computed using our methods.

References

[AE] M. Abrahamsen and S. Eilers. On the asymptotic enumeration of LEGO structures. *Exp. Math.* **20** (2011) 145-152.

[AC] P. H. Algoet and T. Cover. A sandwich proof of the Shannon-McMillan-Breiman theorem. *Ann. Prob.* **16** (1988), 899-909.

[BG] C. Bosbach and W. Gawronski. Strong asymptotics for Jacobi polynomials with varying weights. *Methods Appl. Anal.* **6** (1999) 39-54.

[B] L. Breiman. The individual ergodic theorem of information theory. *Ann. Math. Statist.* **28** (1957) 809-811; correction **31** (1960) 809-810.

[DE1] B. Durhuus and S. Eilers. Enumeration of pyramids of one-dimensional pieces of arbitrary fixed integer length. *21st International Meeting on Probabilistic, Combinatorial and Asymptotic Methods in the Analysis of Algorithms* 143-158, Discrete Math. Theor. Comput. Sci. Proc., AM, *Assoc. Discrete Math. Theor. Comput. Sci., Nancy*, 2010.

[DE2] B. Durhuus and S. Eilers. On the entropy of LEGO. *J. Appl Math. Comput.* **45** (2014) 433-448.

[E] S. Eilers. Personal communication (2018).

[KO] S.F. Khawaja and A.B. Olde Daalhuis. Uniform asymptotic expansions for hypergeometric functions with large parameters IV. *Anal. Appl. (Singap.)* **12** (2014), 667-710.

[JKC] J. Kirk Christiansen. Taljonglering med klodser−eller talrige klodser (Danish). Klodshans. LEGO Company Newsletter (1974).

[KKC] K. Kirk Christiansen. The Ultimate LEGO Book. Dorling Kindersley, London (1999).

[Le] Annual Report 2016, available at `https://www.lego.com/en-us/aboutus/news-room/2017/march/annual-results-2016`. The LEGO Group, 2017.

[Li] D. Lipkowitz. The LEGO Book. Dorling Kindersley, London (2012).

[M] B. McMillan. The basic theorems of information theory. *Ann. Math. Statist.* **24** (1953) 196-219.

[OD] A.B. Olde Daalhuis. Hypergeometric function. *NIST Handbook of Mathematical Functions*, 383-401, U.S. Dept. Commerce, Washington, 2010.

[Sh] C. Shannon. A mathematical theory of communication. *Bell System Technical Journal* **27** (1948), 379-423, 623-656.

[Sl] L. J. Slater. Generalized Hypergeometric Functions. Cambridge University Press, Cambridge (1966).

[Sz] G. Szegö. *Orthogonal Polynomials.* Amer. Math. Soc., Providence, RI, 1975.

Department of Mathematics, Ferris State University, ASC 2021, Big Rapids, MI 49307
E-mail address: `mcclend2@ferris.edu`

2255 Clyde Rd. Howell, MI, 48855
E-mail address: `wilsoj65@ferris.edu`