# Exact Enumeration of Polygon Mosaics

**Jack Hanke**
Northwestern University

**Richard Schank**

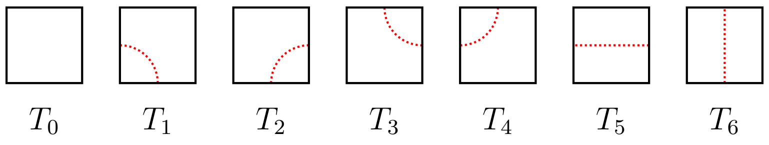
**Michael Maltenfort**
Northwestern University

**Abstract**

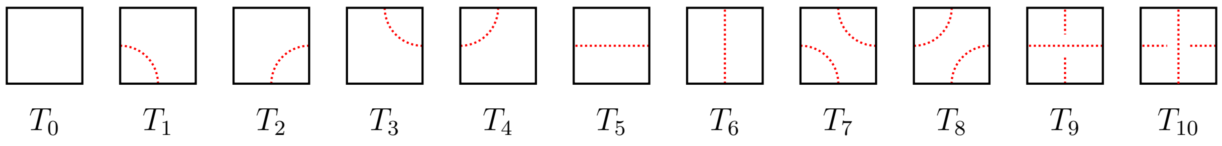
Hong and Oh calculated upper and lower bounds on the number of polygon mosaics for 7 distinct tiles that together model multiple ring polymers in physics. We exactly enumerate these polygon mosaics. The method we introduce generalizes to various other tile sets. We also introduce and enumerate a variation on polygon mosaics called messy polygon mosaics for various tile sets.

## 1 Introduction

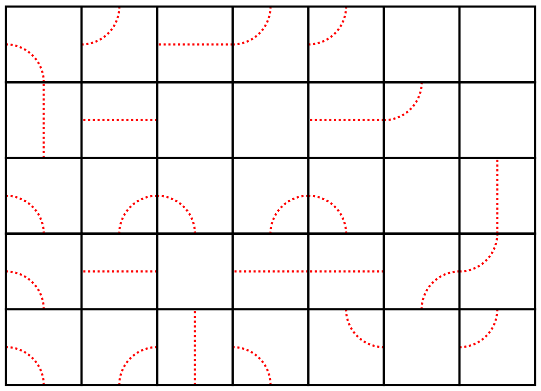
Consider the following set of 7 symbols labeled $\{T_0, \ldots T_6\}$ composed of unit squares and dotted lines connecting pairs of sides at their midpoints.



TODO move following diagram somewhere else



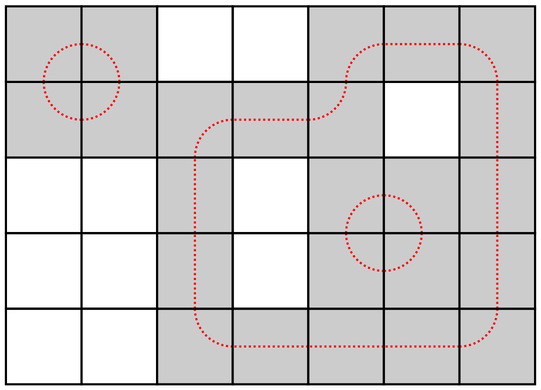
Call these symbols *tiles*. A *mosaic* of size $(m, n)$ is an $m \times n$ matrix of these tiles. For example, below is a mosaic of size $(5, 7)$.



Consider a edge shared between two tiles in the above figure. The edge has either 0, 1, or 2 dotted lines drawn from its midpoint. Also note that the edges of the tiles on the boundary of the matrix are not shared by another tile. Therefore these edges only have 0 or 1 dotted lines drawn from their midpoint. We define a *polygon mosaic* to be a mosaic that has all unit edges having 0 or 2 dotted lines drawn from their midpoint.

We call these polygon mosaics because, other than the mosaic consisting of all $T_1$ tiles, the dotted lines form shapes we call *polygons*[1].

**Example 1.1.** Below is an example of a polygon mosaic of size $(5, 7)$ that contains 3 polygons, with the squares that make them up hilighted in gray.



Note that in large enough mosaics, it is possible for larger polygons to surround smaller polygons.

Let $p_{m,n}$ be the number of polygon mosaics of size $(m, n)$. First notice that if either $m$ or $n$ is 1, one cannot construct a polygon mosaic, so

$$p_{m,1} = p_{1,n} = 0.$$

Hong and Oh [1, Hong2018] gave the following upper and lower bounds[2] for $m, n \geq 2$.

[1]Polygons are more commonly called "self-avoiding polygons" in the literature to emphasize their relationship with self-avoiding walks.

[2]The authors did not consider the mosaic containing all $T_1$ tiles a polygon mosaic, and so define $p_{m,n}$ as one less than what we define.

$$2^{m+n-3}\left(\frac{17}{10}\right)^{(m-2)(n-2)} \le p_{m,n} \le 2^{m+n-3}\left(\frac{31}{16}\right)^{(m-2)(n-2)}.$$

In Theorem 2, we provide an exact expression for $p_{m,n}$ for $m, n \ge 2$. We also introduce messy polygon mosaics—a variant of polygon mosaics—and enumerate them using a similar strategy.

First we define $A(m) \in \mathbb{Z}^{2^{m-1}\times 2^{m-1}}$, which we use to enumerate $p_{m,n}$.

**Definition 1.** First define $A(2) = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$. We recursively define $A(k+1)$ given $A(k)$. Begin by writing $A(k) = \begin{bmatrix} A_{0,0} & A_{0,1} \\ A_{1,0} & A_{1,1} \end{bmatrix}$, where the block matrices $A_{i,j}$ are square block matrices of size $2^{k-2} \times 2^{k-2}$. We then have

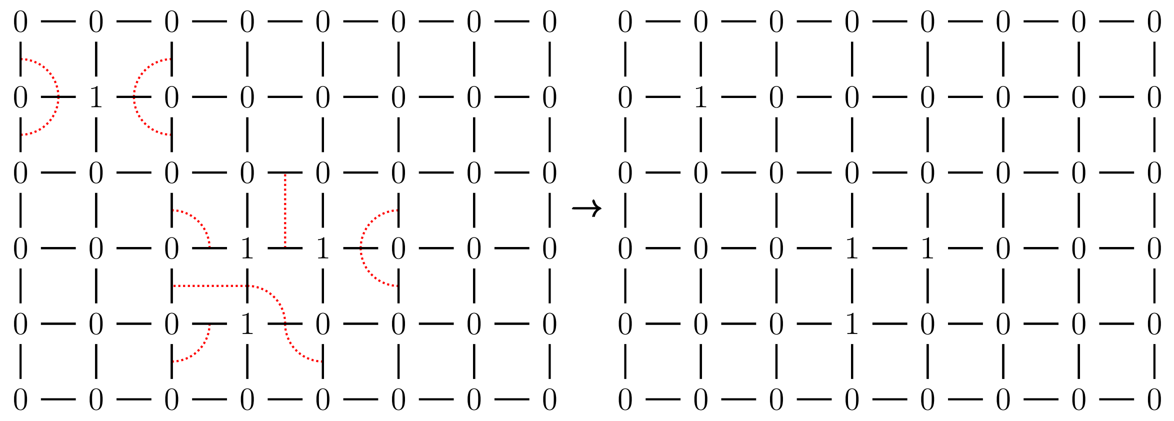
$$A(k+1) = \begin{bmatrix} A_{0,0} & A_{0,0} & A_{0,1} & A_{0,1} \\ A_{0,0} & A_{0,0} & 0A_{0,1} & A_{0,1} \\ A_{1,0} & 0A_{1,0} & A_{1,1} & A_{1,1} \\ A_{1,0} & A_{1,0} & A_{1,1} & A_{1,1} \end{bmatrix},$$

Construct $A(m)$ by starting with $k = 2$ and recursing until $k = m$.

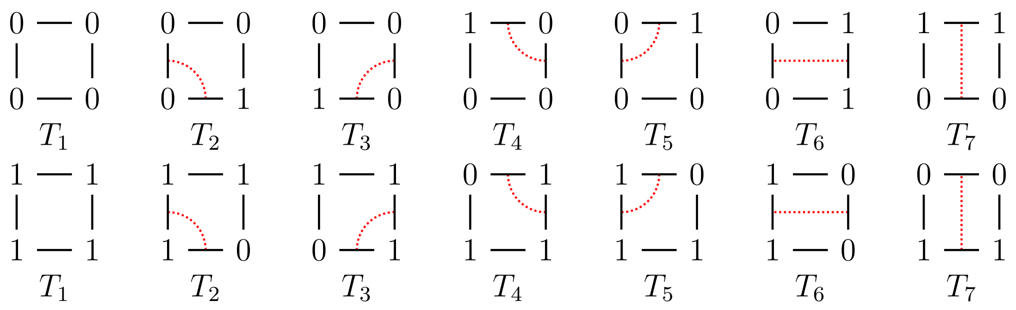
**Theorem 2.** *The number of polygon mosaics $p_{m,n}$ is the $(0,0)$-th entry of $A(m)^n$.*

## 2 Proof of Theorem 2

*Proof.* For a given mosaic, label the vertices of the tiles as follows. If the vertex is surrounded by an even number of polygons, label it 0. If the vertex is surrounded by an odd number of polygons, label it 1. To make a *vertex labeling* we also remove the dotted lines from all tiles in the mosaic. Using the mosaic from Example 1.1, we show both the labeling of the vertices, and then the removal of the dotted lines.

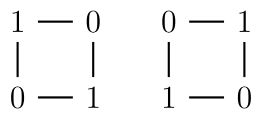


The critical point is this: even with the dotted lines removed, the vertex labeling uniquely identifies the polygon mosaic. This is true even if there are polygons surrounding other polygons. This is because the vertex labeling of an individual cell, which we will call a *cell-labeling*, uniquely corresponds with a cell $T_i$. This is shown below.


$T_1$ $T_2$ $T_3$ $T_4$ $T_5$ $T_6$ $T_7$
$T_1$ $T_2$ $T_3$ $T_4$ $T_5$ $T_6$ $T_7$

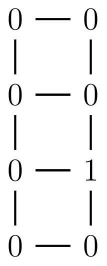
We can then enumerate $p_{m,n}$ by enumerating the number of vertex labelings that correspond with a valid polygon mosaic.

Consider the collection of vertex labelings for a $m \times n$ grid of cells. To correspond with a polygon mosaic, each boundary vertex is necessarily labeled 0, and each interior vertex is labeled 0 or 1. Additionally, of these $2^{(m-1)(n-1)}$ labelings, the labelings that correspond with valid polygon mosaics are ones that do not contain the following two cell-labelings.



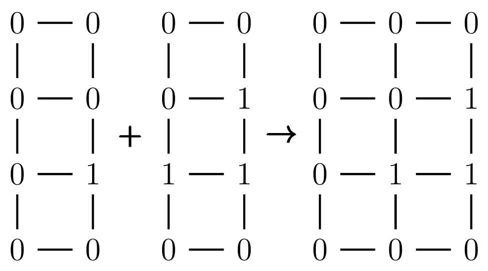
These two cell-labelings aren't associated with any of the cells $T_1, \dots, T_7$, and so cannot correspond with a polygon mosaic. We seek a recursive solution to enumerate the number of vertex labelings for a $m \times n$ grid with these conditions: the labeling has a boundary of all 0's and does not contain the above cell-labelings.

Our solution accomplishes this by building $m \times n$ vertex labelings for fixed $m$ using vertex labelings of $m \times 1$ columns of cells. These columns have vertex labelings such that the top and bottom edge all labeled 0. One of these columns for $m = 3$ is below.



It is useful to define an index for vertex labelings of these $m \times 1$ columns. To do this, consider reading a column of vertex labelings from bottom to top, ignoring the first and last 0's. If we interpret these sequences for the left and right column as binary numbers $b_{left}$ and $b_{right}$, the index in base ten is the pair $(b_{left}, b_{right})$. For example, for the above $m = 3$ column, we have sequences 00 for the left column and 10 for the right column, so the index is $(0, 2)$.

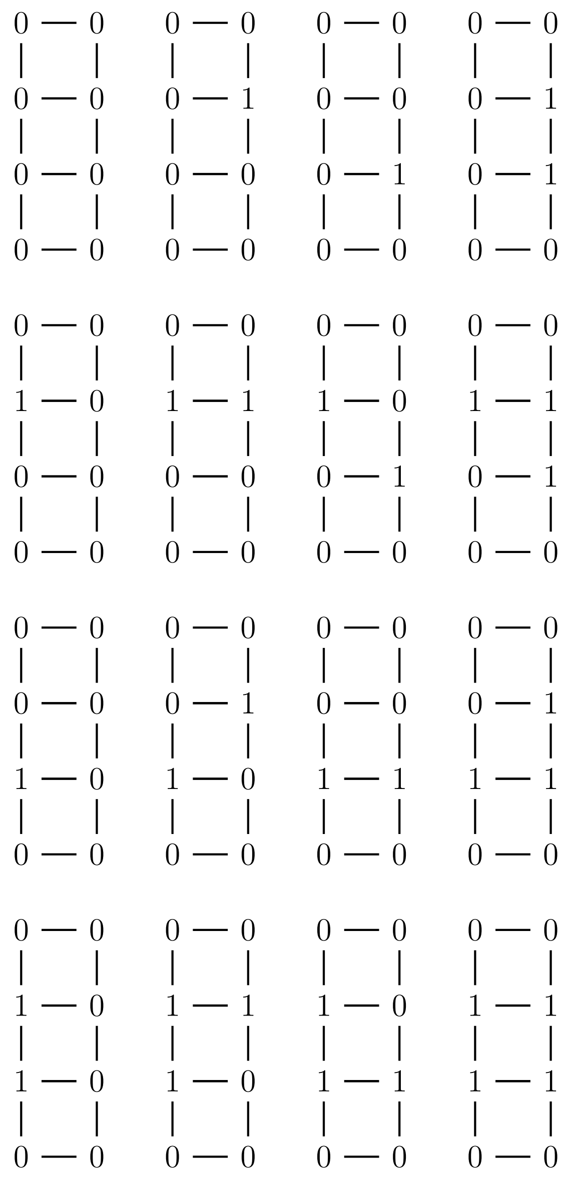
Next consider a column with index $(0, b_1)$ for some $b_1 \in [0, 2^{m-1} - 1]$. For reasons we will see later, let's call this the *starting column*. Now consider appending a column with index $(b_1, b_2)$ for some $b_2 \in [0, 2^{m-1} - 1]$. For example, below is a diagram for appending our starting column $(0, 2)$ with column $(2, 3)$.



Notice that in the above example we have created a vertex labeling for an $m \times 2$ grid of cells, in which the left-most vertex column labels are all 0 (left index of 0). Also note that we did not create either illegal cell-labelings with column $(2, 3)$. However, we could append the column $(2, 1)$, which would create an illegal cell-labeling.

Finally, if we further appended a column with label $(b_2, 0)$, we would have created a vertex labeling with a boundary of all 0's. For this reason, let's call a column with index $(b, 0)$ for $b \in [0, 2^{m-1} - 1]$ an *ending column*. This vertex labeling would correspond with 1 polygon mosaic if the middle column had index $(2, 3)$, but not if the middle column had index $(2, 1)$.

This motivates the creation of a matrix $A(m)$ to every column index $(i, j)$, where $A(m)_{i,j} = 1$ if the column labeling corresponds with a legal polygon mosaic, and 0 otherwise. For example, the $A(3)$ matrix corresponds with the following labeled columns.

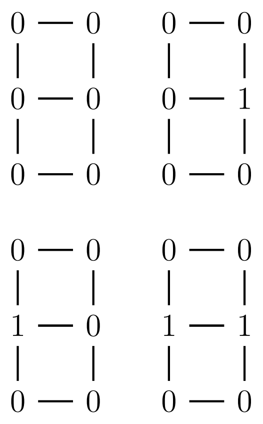


$$\rightarrow \qquad \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

$A(3)$

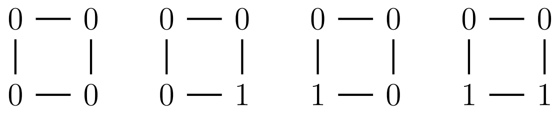
$A(m)$ has the property that the 0-th row represets all starting columns, and the 0-th column represents all ending columns. Even more importantly, notice that $A(m)^2_{i,j}$ represents the number of $m \times 2$ grids with left-most index $i$ and right-most index $j$ that correspond with a legal polygon mosaic. In general, $(A(m)^n)_{i,j}$ represents this quantity for an $m \times n$ grid of cells, and so if we know $A(m)$ for some $m$, then $(A(m)^n)_{0,0} = p_{m,n}$.

The final component of the proof is constructing $A(m)$ for any $m$. Begin by calculating $A(2)$ by identifying the number of legal polygon mosaics that correspond with each vertex coloring index $\{(0,0), (0,1), (1,0), (1,1)\}$, like so.

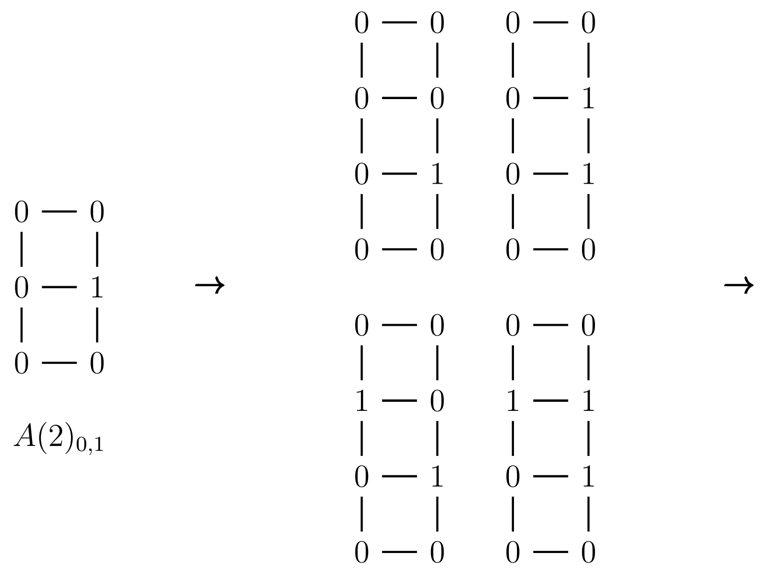


$$\rightarrow \quad \begin{bmatrix}1 & 1\\ 1 & 1\end{bmatrix}$$

Next consider an arbitrary value $A(k)_{i,j}$ for any $k \geq 2$. This value is 1 if the $k \times 1$ column with index $(i,j)$ can be part of a polygon mosaic, and 0 otherwise. We can determine that specific values of $A(k+1)$ are multiples of $A(k)_{i,j}$ by considering the following operation on an arbitrary column with index $(i,j)$. Copy the column four times and replace the top two 0's of each column with the bottom row of labels from one of the four cell-labelings below.



For example, for the $m = 2$ column with index $(0,1)$, this operation looks like the following.



$$\rightarrow \quad \begin{bmatrix}A(3)_{0,2} & A(3)_{0,3}\\ A(3)_{1,2} & A(3)_{1,3}\end{bmatrix}$$

This operation results in 4 new columns that are represented in $A(k+1)$. In our example, specifically we get the following.

$$\begin{bmatrix}A(3)_{0,2} & A(3)_{0,3}\\ A(3)_{1,2} & A(3)_{1,3}\end{bmatrix} = \begin{bmatrix}1A(2)_{i,j} & 1A(2)_{i,j}\\ 0A(2)_{i,j} & 1A(2)_{i,j}\end{bmatrix},$$

Critically, this transformation *only* changes the identity of the top two tiles. This implies that the same value coefficients computed by comparing $A(2)$ and $A(3)$ can be used for any $m \times 1$ column, as long as both column indices $(i, j)$ are congruent mod 2. Furthermore, if one writes $A(k)$ as the block matrix

$$A(k) = \begin{bmatrix} A_{0,0} & A_{0,1} \\ A_{1,0} & A_{1,1} \end{bmatrix},$$

where $A_{\hat{i},\hat{j}} \in \mathbb{R}^{2^{k-2} \times 2^{k-2}}$, then all column's represented in $A_{\hat{i},\hat{j}}$ have indices $(i, j) \equiv (\hat{i}, \hat{j}) \bmod 2$. This allows us to write that in general, if $A(k) = \begin{bmatrix} A_{0,0} & A_{0,1} \\ A_{1,0} & A_{1,1} \end{bmatrix}$, then

$$\begin{bmatrix} V_{0,0}A_{0,0} & V_{0,1}A_{0,0} & V_{0,2}A_{0,1} & V_{0,3}A_{0,1} \\ V_{1,0}A_{0,0} & V_{1,1}A_{0,0} & V_{1,2}A_{0,1} & V_{1,3}A_{0,1} \\ V_{2,0}A_{1,0} & V_{2,1}A_{1,0} & V_{2,2}A_{1,1} & V_{2,3}A_{1,1} \\ V_{3,0}A_{1,0} & V_{3,1}A_{1,0} & V_{3,2}A_{1,1} & V_{3,3}A_{1,1} \end{bmatrix}. \tag{1}$$

where $V \in \mathbb{R}^{4 \times 4}$ can be found after directly computing $A(2)$ and $A(3)$, then solving the following equation.

$$\begin{bmatrix} A(3)_{0,0} & A(3)_{0,1} & A(3)_{0,2} & A(3)_{0,3} \\ A(3)_{1,0} & A(3)_{1,1} & A(3)_{1,2} & A(3)_{1,3} \\ A(3)_{2,0} & A(3)_{2,1} & A(3)_{2,2} & A(3)_{2,3} \\ A(3)_{3,0} & A(3)_{3,1} & A(3)_{3,2} & A(3)_{3,3} \end{bmatrix} = \begin{bmatrix} V_{0,0}A(2)_{0,0} & V_{0,1}A(2)_{0,0} & V_{0,2}A(2)_{0,1} & V_{0,3}A(2)_{0,1} \\ V_{1,0}A(2)_{0,0} & V_{1,1}A(2)_{0,0} & V_{1,2}A(2)_{0,1} & V_{1,3}A(2)_{0,1} \\ V_{2,0}A(2)_{1,0} & V_{2,1}A(2)_{1,0} & V_{2,2}A(2)_{1,1} & V_{2,3}A(2)_{1,1} \\ V_{3,0}A(2)_{1,0} & V_{3,1}A(2)_{1,0} & V_{3,2}A(2)_{1,1} & V_{3,3}A(2)_{1,1} \end{bmatrix}$$

This is solved by

$$V = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

which completes the proof.

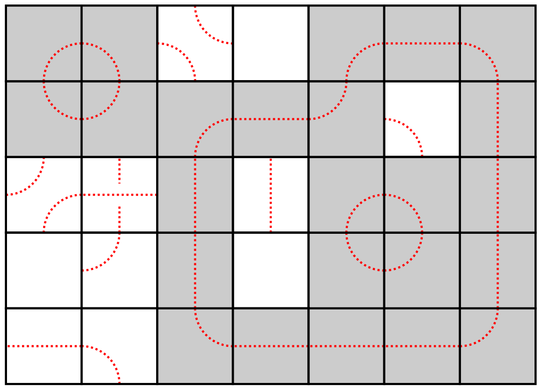
□

The method detailed in Theorem 2 generalizes to other tile sets, which we tabularize in Section 5 without proof. Interestingly, the method not only generalizes to other tile sets, but can also be augmented to enumerate the more complicated "messy" polygon mosaics.

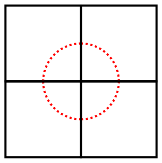
## 3 Messy Polygon Mosaics

A *messy polygon mosaic* is a mosaic that contains at least one polygon, with no restriction on other shared edges.

**Example 3.1.** Below is a messy polygon mosaic of size $(5, 7)$ that contains 1 polygon, with the squares that make it up hilighted in gray.



It turns out that it is simpler to enumerate the number of mosaics that *do not* contain a polygon. Therefore, let $t_{m,n}$ be the number of mosaics that do not contain a polygon. Also from the fact that the smallest polygon is



we have that $t_{n,1} = 7^n$, and $t_{2,2} = 7^4 - 1$. We can use similar techniques to Theorem 2 to enumerate messy polygon mosaics. We define $A(m) \in \mathbb{Z}^{2^{m-1} \times 2^{m-1}}$, which we use to enumerate $t_{m,n}$.

**Definition 3.** First define $A(2) = \begin{bmatrix} 7^2 & 1 \\ -1 & 1 \end{bmatrix}$. We recursively define $A(k+1) \in \mathbb{Z}^{2^{m-1} \times 2^{m-1}}$ given $A(k)$. Begin by writing $A(k) = \begin{bmatrix} A_{0,0} & A_{0,1} \\ A_{1,0} & A_{1,1} \end{bmatrix}$, where the block matrices $A_{i,j}$ are square block matrices of size $2^{k-2} \times 2^{k-2}$. We then have

$$A(k+1) = \begin{bmatrix} 7A_{0,0} & \frac{1}{7}A_{0,0} & 7A_{0,1} & A_{0,1} \\ -\frac{1}{7}A_{0,0} & A_{0,0} & 0A_{0,1} & A_{0,1} \\ 7A_{1,0} & 0A_{1,0} & 7A_{1,1} & A_{1,1} \\ A_{1,0} & -A_{1,0} & A_{1,1} & 7A_{1,1} \end{bmatrix}.$$

Construct $A(m)$ by starting with $k = 2$ and recursing until $k = m$.

**Theorem 4.** *The number of mosaics that do not contain a polygon is the* $(0,0)$*-th entry of* $A(m)^n$.

## 4 Proof of Theorem 4

*Proof.* We consider the same vertex labeling in the proof of Theorem 2, namely labeling a vertex by the number of polygons that contain it mod 2. Though avoided in the previous proof for clarity, it is important to think of each individual cell-labeling as having its own weight, which corresponds with how many distinct tiles can map to the cell-labeling.

$$\begin{array}{cccccccc}
0 - 0 & 0 - 0 & 0 - 0 & 0 - 0 & 0 - 1 & 0 - 1 & 0 - 1 & 0 - 1 \\
\mid \quad \mid & \mid \quad \mid & \mid \quad \mid & \mid \quad \mid & \mid \quad \mid & \mid \quad \mid & \mid \quad \mid & \mid \quad \mid \\
0 - 0 & 0 - 1 & 1 - 0 & 1 - 1 & 0 - 0 & 0 - 1 & 1 - 0 & 1 - 1 \\
w_1 & w_2 & w_3 & w_4 & w_5 & w_6 & w_7 & w_8 \\
1 - 0 & 1 - 0 & 1 - 0 & 1 - 0 & 1 - 1 & 1 - 1 & 1 - 1 & 1 - 1 \\
\mid \quad \mid & \mid \quad \mid & \mid \quad \mid & \mid \quad \mid & \mid \quad \mid & \mid \quad \mid & \mid \quad \mid & \mid \quad \mid \\
0 - 0 & 0 - 1 & 1 - 0 & 1 - 1 & 0 - 0 & 0 - 1 & 1 - 0 & 1 - 1 \\
w_9 & w_{10} & w_{11} & w_{12} & w_{13} & w_{14} & w_{15} & w_{16}
\end{array}$$

The proof of Theorem 2 can be seen as assigning $w_7 = w_{10} = 0$, and all other weights to 1. The usefulness of this view can be seen when considering the operation for creating the recursive definition for $A(k)$ in the previous proof. Previously, we defined the coefficient matrix $V$ by computing $A(2)$ and $A(3)$ directly and comparing. Now with a weight assigned to individual cell-labelings, we can define the values of $A(2)$ and $V$ directly in terms of these weights.

$$A(2) = \begin{bmatrix} w_1w_1 & w_2w_5 \\ w_3w_9 & w_4w_{13} \end{bmatrix}, V = \begin{bmatrix} \frac{w_1w_1}{w_1} & \frac{w_2w_5}{w_1} & \frac{w_1w_2}{w_2} & \frac{w_2w_6}{w_2} \\ \frac{w_3w_9}{w_1} & \frac{w_4w_{13}}{w_1} & \frac{w_3w_{10}}{w_2} & \frac{w_4w_{14}}{w_2} \\ \frac{w_1w_3}{w_3} & \frac{w_2w_7}{w_3} & \frac{w_1w_4}{w_4} & \frac{w_2w_8}{w_4} \\ \frac{w_3w_{11}}{w_3} & \frac{w_4w_{15}}{w_3} & \frac{w_3w_{12}}{w_4} & \frac{w_4w_{16}}{w_4} \end{bmatrix} = \begin{bmatrix} w_1 & \frac{w_2w_5}{w_1} & w_1 & w_6 \\ \frac{w_3w_9}{w_1} & \frac{w_4w_{13}}{w_1} & \frac{w_3w_{10}}{w_2} & \frac{w_4w_{14}}{w_2} \\ w_1 & \frac{w_2w_7}{w_3} & w_1 & \frac{w_2w_8}{w_4} \\ w_{11} & \frac{w_4w_{15}}{w_3} & \frac{w_3w_{12}}{w_4} & w_{16} \end{bmatrix}.$$

With this identity, we can enumerate $t_{m,n}$ once we have proper assignments for the 16 weights. As in the previous proof, we have $w_7 = w_{10} = 0$, as again these are impossible vertex labelings for our tile set.

Next consider the cell labellings for $w_1$ and $w_{16}$. When enumerating polygon mosaics (and their messy variant), these cell-labelings do not contribute to the cells of a polygon. For polygon mosaics, only the $T_1$ tile are permitted to not contribute to the shape of a polygon. However, in messy polygon mosaics, all 7 tiles are permitted to not contribute to the shape of the polygon, so $w_1 = w_{16} = 7$.

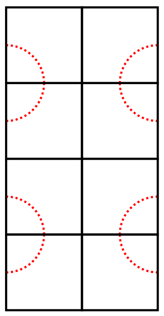
However, this means we now lose the uniqueness of the map from vertex labeling to messy polygon mosaics. For instance, the sub-grid vertex labelings below are now ambiguous as to whether or not they represent a polygon.

$$\begin{array}{ccc}
0 - 0 - 0 & \quad & 1 - 1 - 1 \\
\mid \quad \mid \quad \mid & & \mid \quad \mid \quad \mid \\
0 - 0 - 0 & & 1 - 1 - 1 \\
\mid \quad \mid \quad \mid & & \mid \quad \mid \quad \mid \\
0 - 0 - 0 & & 1 - 1 - 1
\end{array}$$

This ambiguity is explored in the following example.

**Example 4.1.** Consider the four vertex labelings below, along with the messy polygon mosaic on the right. Write the product of the weights of all cell-labellings below each, assuming all weights not defined above are 1.

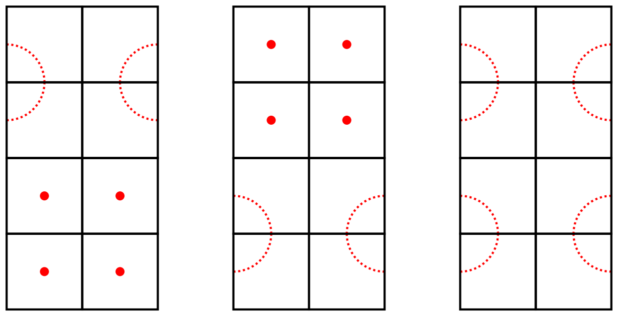
$$\begin{array}{ccccc} 0 & — & 0 & — & 0 \\ | & & | & & | \\ 0 & — & 0 & — & 0 \\ | & & | & & | \\ 0 & — & 0 & — & 0 \\ | & & | & & | \\ 0 & — & 0 & — & 0 \\ | & & | & & | \\ 0 & — & 0 & — & 0 \end{array} \qquad \begin{array}{ccccc} 0 & — & 0 & — & 0 \\ | & & | & & | \\ 0 & — & 1 & — & 0 \\ | & & | & & | \\ 0 & — & 0 & — & 0 \\ | & & | & & | \\ 0 & — & 0 & — & 0 \\ | & & | & & | \\ 0 & — & 0 & — & 0 \end{array} \qquad \begin{array}{ccccc} 0 & — & 0 & — & 0 \\ | & & | & & | \\ 0 & — & 0 & — & 0 \\ | & & | & & | \\ 0 & — & 0 & — & 0 \\ | & & | & & | \\ 0 & — & 1 & — & 0 \\ | & & | & & | \\ 0 & — & 0 & — & 0 \end{array} \qquad \begin{array}{ccccc} 0 & — & 0 & — & 0 \\ | & & | & & | \\ 0 & — & 1 & — & 0 \\ | & & | & & | \\ 0 & — & 0 & — & 0 \\ | & & | & & | \\ 0 & — & 1 & — & 0 \\ | & & | & & | \\ 0 & — & 0 & — & 0 \end{array}$$



$$7^8 \qquad\qquad 7^4 \qquad\qquad 7^4 \qquad\qquad 7^0$$

Notice that each vertex labelling could include the right-most messy polygon mosaic. In fact, the left-most vertex labeling contains all possible mosaics! If we were to add these weight products together, we would count the right messy polygon mosaic 4 times.

The double counting demonstrated in Example 4.1 motivates the following idea. If vertex labellings with an odd number of polygons are negative, then the addition of these weight products would incorporate the *inclusion-exclusion principle*, mitigating the double counting. In Example 4.1, the sum $7^8 - 7^4 - 7^4 + 7^0$ would then represent the number of mosaics that *do not* contain the following three classes of messy polygon mosaics, where cells that can be any tile are marked with a dot.



Therefore, the sum over the products for all vertex labelings, where the product is negative if the vertex labeling represents an odd number of polygons in the mosaic, would be the number of mosaics that *do not* include messy polygon mosaics.

We can accomplish this by finding a weight assignment such that the product over the cell-labeling weights of *any* single polygon equals $-1$. It is not obvious that such an assignment can even be found!

Luckily such assignments exist. The proof of this fact can be found in the Appendix, and choosing an assignment gives us the following weight assignments.

|  |  |  |  |  |  |  |  |
|---|---|---|---|---|---|---|---|
| 0 — 0 | 0 — 0 | 0 — 0 | 0 — 0 | 0 — 1 | 0 — 1 | 0 — 1 | 0 — 1 |
| \| \| | \| \| | \| \| | \| \| | \| \| | \| \| | \| \| | \| \| |
| 0 — 0 | 0 — 1 | 1 — 0 | 1 — 1 | 0 — 0 | 0 — 1 | 1 — 0 | 1 — 1 |
| $w_1 = 7$ | $w_2 = 1$ | $w_3 = 1$ | $w_4 = 1$ | $w_5 = 1$ | $w_6 = 1$ | $w_7 = 0$ | $w_8 = 1$ |
| 1 — 0 | 1 — 0 | 1 — 0 | 1 — 0 | 1 — 1 | 1 — 1 | 1 — 1 | 1 — 1 |
| \| \| | \| \| | \| \| | \| \| | \| \| | \| \| | \| \| | \| \| |
| 0 — 0 | 0 — 1 | 1 — 0 | 1 — 1 | 0 — 0 | 0 — 1 | 1 — 0 | 1 — 1 |
| $w_9 = -1$ | $w_{10} = 0$ | $w_{11} = 1$ | $w_{12} = 1$ | $w_{13} = 1$ | $w_{14} = 1$ | $w_{15} = -1$ | $w_{16} = 7$ |

This immediately gives us a way to construct an analagous definition for $A(k+1)$ given $A(k)$. Once we write $A(k) = \begin{bmatrix} A_{0,0} & A_{0,1} \\ A_{1,0} & A_{1,1} \end{bmatrix}$, we have

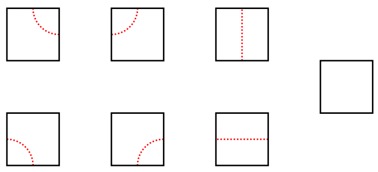
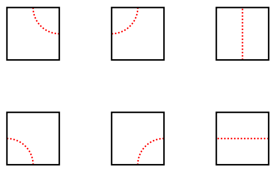
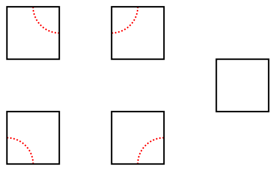
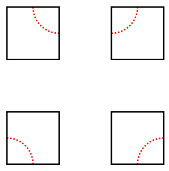
$$A(2) = \begin{bmatrix} 7^2 & 1 \\ -1 & 1 \end{bmatrix}, V = \begin{bmatrix} 7 & \frac{1}{7} & 7 & 1 \\ -\frac{1}{7} & 1 & 0 & 1 \\ 7 & 0 & 7 & 1 \\ 1 & -1 & 1 & 7 \end{bmatrix}$$

Subsituting $V$ into Equation 1 gives the result.

□

## 5 Summary of Results

As demonstrated in the proof of Theorem 4, the enumeration of both polygon mosaics and mosaics that do not contain a polygon share the same structure, and only differ by the identity of the matrices $A(2), V$. We summarize these matrices for various collections of tiles below.

| Tile Set | Polygon Mosaics | Messy Polygon Mosaics |
|---|---|---|
|  | $\begin{bmatrix}1 & 1\\1 & 1\end{bmatrix}, \begin{bmatrix}1 & 1 & 1 & 1\\1 & 1 & 0 & 1\\1 & 0 & 1 & 1\\1 & 1 & 1 & 1\end{bmatrix}$ | $\begin{bmatrix}7^2 & 1\\-1 & 1\end{bmatrix}, \begin{bmatrix}7 & \frac{1}{7} & 7 & 1\\-\frac{1}{7} & 1 & 0 & 1\\7 & 0 & 7 & 1\\1 & -1 & 1 & 7\end{bmatrix}$ |
|  | TODO | $\begin{bmatrix}6^2 & 1\\-1 & 1\end{bmatrix}, \begin{bmatrix}6 & \frac{1}{6} & 6 & 1\\-\frac{1}{6} & 1 & 0 & 1\\6 & 0 & 6 & 1\\1 & -1 & 1 & 6\end{bmatrix}$ |
|  | $\begin{bmatrix}1 & 1\\1 & 0\end{bmatrix}$, TODO | TODO |
|  | TODO | TODO |

## References

[1] Kyungpyo Hong and Seungsang Oh. "Bounds on Multiple Self-avoiding Polygons". In: *Canadian Mathematical Bulletin* 61.3 (Sept. 2018), pp. 518–530. ISSN: 1496-4287. DOI: 10.4153/cmb-2017-072-x. URL: http://dx.doi.org/10.4153/CMB-2017-072-x.

## 6 Appendix

We demonstrate that weight assignments exist such that the product over the cell-labeling weights of *all* single polygons equals $-1$. We do this by first asserting that the product of the weights associated with the smallest polygon multiply to $-1$, ie. $w_2 w_3 w_5 w_9 = -1$.

**Lemma 5.** *One can construct all larger polygons from the smallest polygon using a finite set of transformations $S$.*

*Proof.* TODO Something about chaging vertex values □

This is because one can find $w_1, \ldots, w_{16}$ so that the following two constraints hold:

**Constraint 6.** *The weights associated with the smallest polygon multiply to $-1$, ie.* $w_2 w_3 w_5 w_9 = -1$.

**Constraint 7.** *All transformations in $S$ preserve the weight product of a changed polygon.*

Constraint 6 and Constraint 7 amount to a series of constraints on the values of $w_i$. Choosing a solution set from these constraints gives the following weights.

Flipping the parity of a single vertex in a vertex labeling changes the 4 surrounding cells. This creates a constraint on a subset of $w_1, \ldots, w_{16}$.

The flipping of parity of a single vertex results in 2 distinct types of constraints. Let a constraint of *Type 1* be a parity flip that does not change the number of polygons represented in the vertex labeling. For example, consider the following flip of the center vertex in the following sub vertex labeling.

$$\begin{array}{ccccccccccccc}
0 & — & 0 & — & 0 & & 0 & — & 0 & — & 0 \\
| & & | & & | & & | & & | & & | \\
0 & — & 0 & — & 1 & \rightarrow & 0 & — & 1 & — & 1 \\
| & & | & & | & & | & & | & & | \\
0 & — & 1 & — & 1 & & 0 & — & 1 & — & 1
\end{array}$$

As this does not change the associated number of polygons in the larger vertex labeling, we want this to preserve the sign of the weight product. This gives the following associated constraint.

$$\operatorname{sign}(w_1 w_2 w_5 w_9) = \operatorname{sign}(w_2 w_4 w_6 w_{16}).$$

Now let a constraint of *Type 2* be a parity flip that does change the number of polygons. For example, consider flipping the center vertex of the following portion of a vertex labeling.

$$\begin{array}{ccccccccccccc}
0 & — & 0 & — & 0 & & 0 & — & 0 & — & 0 \\
| & & | & & | & & | & & | & & | \\
1 & — & 0 & — & 1 & \rightarrow & 1 & — & 1 & — & 1 \\
| & & | & & | & & | & & | & & | \\
0 & — & 0 & — & 0 & & 0 & — & 0 & — & 0
\end{array}$$

The above transformation corresponds with *either* two distinct polygons joining into one polygon *or* one polygon splitting into two distinct polygons. In either case, we want the sign of the product to switch. This corresponds with the following constraint.

$$\operatorname{sign}(w_3 w_2 w_9 w_5) = -\operatorname{sign}(w_4 w_4 w_{13} w_{13}).$$

All Type 1 constraints are as follows. For the following set of equations, assume the equals sign (=) means *only* equal in sign.

$$w_1w_1w_2w_3 = w_2w_3w_6w_{11} \qquad w_1w_1w_2w_4 = w_2w_3w_6w_{12} \qquad w_1w_1w_4w_3 = w_2w_3w_8w_{11}$$
$$w_1w_1w_4w_4 = w_2w_3w_8w_{12} \qquad w_1w_2w_1w_5 = w_2w_4w_5w_{13} \qquad w_1w_2w_1w_6 = w_2w_4w_5w_{14}$$
$$w_1w_2w_2w_8 = w_2w_4w_6w_{16} \qquad w_1w_2w_4w_8 = w_2w_4w_8w_{16} \qquad w_3w_1w_9w_1 = w_4w_3w_{13}w_9$$
$$w_3w_1w_{11}w_1 = w_4w_3w_{15}w_9 \qquad w_3w_1w_{12}w_3 = w_4w_3w_{16}w_{11} \qquad w_3w_1w_{12}w_4 = w_4w_3w_{16}w_{12}$$
$$w_3w_2w_{12}w_8 = w_4w_4w_{16}w_{16} \qquad w_1w_6w_1w_5 = w_2w_8w_5w_{13} \qquad w_1w_6w_1w_6 = w_2w_8w_5w_{14}$$
$$w_1w_6w_2w_8 = w_2w_8w_6w_{16} \qquad w_1w_6w_4w_8 = w_2w_8w_8w_{16} \qquad w_3w_6w_{12}w_8 = w_4w_8w_{16}w_{16}$$
$$w_5w_9w_1w_1 = w_6w_{11}w_5w_9 \qquad w_5w_{13}w_1w_1 = w_6w_{15}w_5w_9 \qquad w_5w_{14}w_1w_5 = w_6w_{16}w_5w_{13}$$
$$w_5w_{14}w_1w_6 = w_6w_{16}w_5w_{14} \qquad w_5w_{14}w_2w_8 = w_6w_{16}w_6w_{16} \qquad w_5w_{14}w_4w_8 = w_6w_{16}w_8w_{16}$$
$$w_{11}w_1w_9w_1 = w_{12}w_3w_{13}w_9 \qquad w_{11}w_1w_{11}w_1 = w_{12}w_3w_{15}w_9 \qquad w_{11}w_1w_{12}w_3 = w_{12}w_3w_{16}w_{11}$$
$$w_{11}w_1w_{12}w_4 = w_{12}w_3w_{16}w_{12} \qquad w_{11}w_2w_{12}w_8 = w_{12}w_4w_{16}w_{16} \qquad w_{11}w_6w_{12}w_8 = w_{12}w_8w_{16}w_{16}$$
$$w_{13}w_9w_1w_1 = w_{14}w_{11}w_5w_9 \qquad w_{15}w_9w_9w_1 = w_{16}w_{11}w_{13}w_9 \qquad w_{15}w_9w_{11}w_1 = w_{16}w_{11}w_{15}w_9$$
$$w_{15}w_9w_{12}w_3 = w_{16}w_{11}w_{16}w_{11} \qquad w_{15}w_9w_{12}w_4 = w_{16}w_{11}w_{16}w_{12} \qquad w_{13}w_{13}w_1w_1 = w_{14}w_{15}w_5w_9$$
$$w_{13}w_{14}w_1w_5 = w_{14}w_{16}w_5w_{13} \qquad w_{13}w_{14}w_1w_6 = w_{14}w_{16}w_5w_{14} \qquad w_{13}w_{14}w_2w_8 = w_{14}w_{16}w_6w_{16}$$
$$w_{13}w_{14}w_4w_8 = w_{14}w_{16}w_8w_{16} \qquad w_{15}w_{13}w_9w_1 = w_{16}w_{15}w_{13}w_9 \qquad w_{15}w_{13}w_{11}w_1 = w_{16}w_{15}w_{15}w_9$$
$$w_{15}w_{13}w_{12}w_3 = w_{16}w_{15}w_{16}w_{11} \qquad w_{15}w_{13}w_{12}w_4 = w_{16}w_{15}w_{16}w_{12} \qquad w_{15}w_{14}w_9w_5 = w_{16}w_{16}w_{13}w_{13}$$
$$w_{15}w_{14}w_9w_6 = w_{16}w_{16}w_{13}w_{14} \qquad w_{15}w_{14}w_{11}w_5 = w_{16}w_{16}w_{15}w_{13} \qquad w_{15}w_{14}w_{11}w_6 = w_{16}w_{16}w_{15}w_{14}$$

Similarly, all Type 2 constraints are as follows. Again, for the following set of equations, assume the equals sign (=) means *only* equal in sign.

$$-w_3w_2w_9w_5 = w_4w_4w_{13}w_{13} \qquad -w_3w_2w_9w_6 = w_4w_4w_{13}w_{14} \qquad -w_3w_2w_{11}w_5 = w_4w_4w_{15}w_{13}$$
$$-w_3w_2w_{11}w_6 = w_4w_4w_{15}w_{14} \qquad -w_3w_6w_9w_5 = w_4w_8w_{13}w_{13} \qquad -w_3w_6w_9w_6 = w_4w_8w_{13}w_{14}$$
$$-w_3w_6w_{11}w_5 = w_4w_8w_{15}w_{13} \qquad -w_3w_6w_{11}w_6 = w_4w_8w_{15}w_{14} \qquad -w_5w_9w_2w_3 = w_6w_{11}w_6w_{11}$$
$$-w_5w_9w_2w_4 = w_6w_{11}w_6w_{12} \qquad -w_5w_9w_4w_3 = w_6w_{11}w_8w_{11} \qquad -w_5w_9w_4w_4 = w_6w_{11}w_8w_{12}$$
$$-w_5w_{13}w_2w_3 = w_6w_{15}w_6w_{11} \qquad -w_5w_{13}w_2w_4 = w_6w_{15}w_6w_{12} \qquad -w_5w_{13}w_4w_3 = w_6w_{15}w_8w_{11}$$
$$-w_5w_{13}w_4w_4 = w_6w_{15}w_8w_{12} \qquad -w_{11}w_2w_9w_5 = w_{12}w_4w_{13}w_{13} \qquad -w_{11}w_2w_9w_6 = w_{12}w_4w_{13}w_{14}$$
$$-w_{11}w_2w_{11}w_5 = w_{12}w_4w_{15}w_{13} \qquad -w_{11}w_2w_{11}w_6 = w_{12}w_4w_{15}w_{14} \qquad -w_{11}w_6w_9w_5 = w_{12}w_8w_{13}w_{13}$$
$$-w_{11}w_6w_9w_6 = w_{12}w_8w_{13}w_{14} \qquad -w_{11}w_6w_{11}w_5 = w_{12}w_8w_{15}w_{13} \qquad -w_{11}w_6w_{11}w_6 = w_{12}w_8w_{15}w_{14}$$
$$-w_{13}w_9w_2w_3 = w_{14}w_{11}w_6w_{11} \qquad -w_{13}w_9w_2w_4 = w_{14}w_{11}w_6w_{12} \qquad -w_{13}w_9w_4w_3 = w_{14}w_{11}w_8w_{11}$$
$$-w_{13}w_9w_4w_4 = w_{14}w_{11}w_8w_{12} \qquad -w_{13}w_{13}w_2w_3 = w_{14}w_{15}w_6w_{11} \qquad -w_{13}w_{13}w_2w_4 = w_{14}w_{15}w_6w_{12}$$
$$-w_{13}w_{13}w_4w_3 = w_{14}w_{15}w_8w_{11} \qquad -w_{13}w_{13}w_4w_4 = w_{14}w_{15}w_8w_{12}$$

Solving all Type 1 and Type 2 constraints gives the following solution set.

| $w_1$ | $w_2$ | $w_3$ | $w_4$ | $w_5$ | $w_6$ | $w_7$ | $w_8$ | $w_9$ | $w_{10}$ | $w_{11}$ | $w_{12}$ | $w_{13}$ | $w_{14}$ | $w_{15}$ | $w_{16}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 7 | -1 | -1 | -1 | -1 | -1 | 0 | -1 | 1 | 0 | -1 | -1 | -1 | -1 | 1 | 7 |
| 7 | -1 | -1 | -1 | -1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | -1 | 1 | -1 | 7 |
| 7 | -1 | -1 | -1 | 1 | -1 | 0 | -1 | -1 | 0 | -1 | -1 | -1 | 1 | -1 | 7 |
| 7 | -1 | -1 | -1 | 1 | 1 | 0 | 1 | -1 | 0 | 1 | 1 | -1 | -1 | 1 | 7 |
| 7 | -1 | -1 | 1 | -1 | -1 | 0 | 1 | 1 | 0 | -1 | 1 | 1 | 1 | -1 | 7 |
| 7 | -1 | -1 | 1 | -1 | 1 | 0 | -1 | 1 | 0 | 1 | -1 | 1 | -1 | 1 | 7 |
| 7 | -1 | -1 | 1 | 1 | -1 | 0 | 1 | -1 | 0 | -1 | 1 | 1 | -1 | 1 | 7 |
| 7 | -1 | -1 | 1 | 1 | 1 | 0 | -1 | -1 | 0 | 1 | -1 | 1 | 1 | -1 | 7 |
| 7 | -1 | 1 | -1 | -1 | -1 | 0 | -1 | -1 | 0 | -1 | 1 | -1 | -1 | -1 | 7 |
| 7 | -1 | 1 | -1 | -1 | 1 | 0 | 1 | -1 | 0 | 1 | -1 | -1 | 1 | 1 | 7 |
| 7 | -1 | 1 | -1 | 1 | -1 | 0 | -1 | 1 | 0 | -1 | 1 | -1 | 1 | 1 | 7 |
| 7 | -1 | 1 | -1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | -1 | -1 | -1 | -1 | 7 |
| 7 | -1 | 1 | 1 | -1 | -1 | 0 | 1 | -1 | 0 | -1 | -1 | 1 | 1 | 1 | 7 |
| 7 | -1 | 1 | 1 | -1 | 1 | 0 | -1 | -1 | 0 | 1 | 1 | 1 | -1 | -1 | 7 |
| 7 | -1 | 1 | 1 | 1 | -1 | 0 | 1 | 1 | 0 | -1 | -1 | 1 | -1 | -1 | 7 |
| 7 | -1 | 1 | 1 | 1 | 1 | 0 | -1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 7 |
| 7 | 1 | -1 | -1 | -1 | -1 | 0 | 1 | -1 | 0 | -1 | -1 | -1 | -1 | -1 | 7 |
| 7 | 1 | -1 | -1 | -1 | 1 | 0 | -1 | -1 | 0 | 1 | 1 | -1 | 1 | 1 | 7 |
| 7 | 1 | -1 | -1 | 1 | -1 | 0 | 1 | 1 | 0 | -1 | -1 | -1 | 1 | 1 | 7 |
| 7 | 1 | -1 | -1 | 1 | 1 | 0 | -1 | 1 | 0 | 1 | 1 | -1 | -1 | -1 | 7 |
| 7 | 1 | -1 | 1 | -1 | -1 | 0 | -1 | -1 | 0 | -1 | 1 | 1 | 1 | 1 | 7 |
| 7 | 1 | -1 | 1 | -1 | 1 | 0 | 1 | -1 | 0 | 1 | -1 | 1 | -1 | -1 | 7 |
| 7 | 1 | -1 | 1 | 1 | -1 | 0 | -1 | 1 | 0 | -1 | 1 | 1 | -1 | -1 | 7 |
| 7 | 1 | -1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | -1 | 1 | 1 | 1 | 7 |
| 7 | 1 | 1 | -1 | -1 | -1 | 0 | 1 | 1 | 0 | -1 | 1 | -1 | -1 | 1 | 7 |
| 7 | 1 | 1 | -1 | -1 | 1 | 0 | -1 | 1 | 0 | 1 | -1 | -1 | 1 | -1 | 7 |
| 7 | 1 | 1 | -1 | 1 | -1 | 0 | 1 | -1 | 0 | -1 | 1 | -1 | 1 | -1 | 7 |
| 7 | 1 | 1 | -1 | 1 | 1 | 0 | -1 | -1 | 0 | 1 | -1 | -1 | -1 | 1 | 7 |
| 7 | 1 | 1 | 1 | -1 | -1 | 0 | -1 | 1 | 0 | -1 | -1 | 1 | 1 | -1 | 7 |
| 7 | 1 | 1 | 1 | -1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | -1 | 1 | 7 |
| 7 | 1 | 1 | 1 | 1 | -1 | 0 | -1 | -1 | 0 | -1 | -1 | 1 | -1 | 1 | 7 |
| 7 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | -1 | 0 | 1 | 1 | 1 | 1 | -1 | 7 |

Any of these assignments are sufficient for calculating $t_{m,n}$.