Tromino Tiling

Liu Hang, Surya Mathialagan, Sean Pek Yu-Xuan

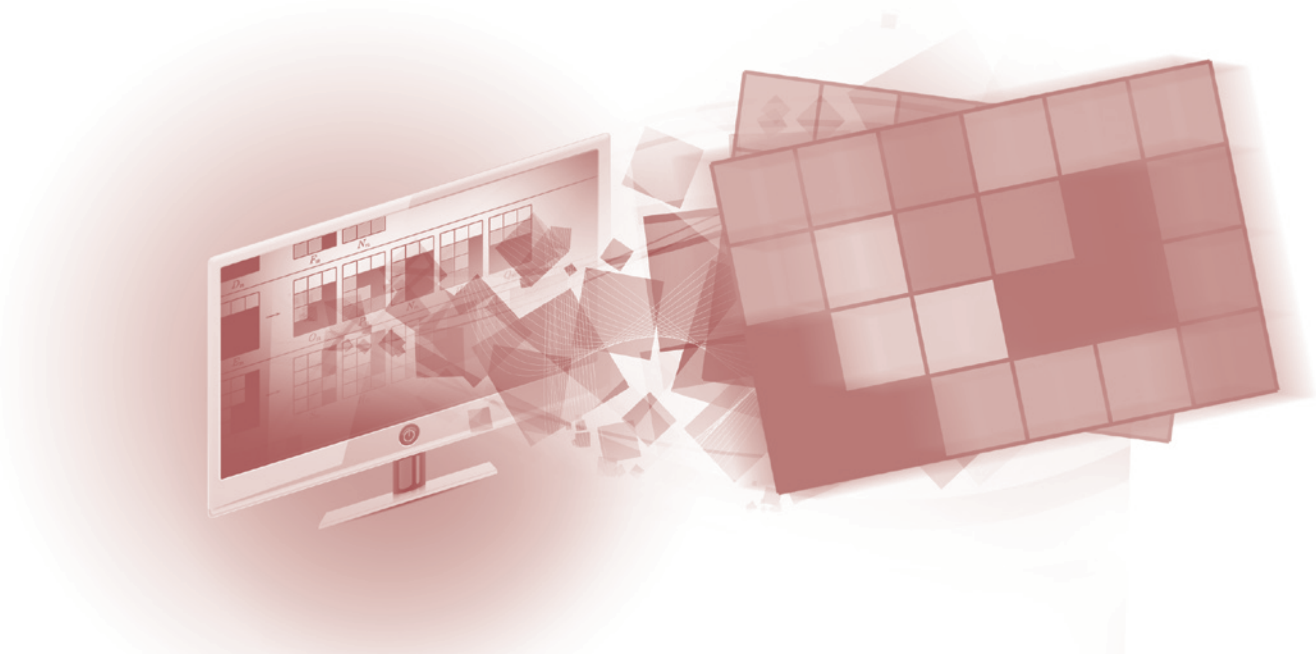
Winner of Foo Kean Pew Memorial Prize and Gold Award of Singapore Mathematical Project Festival 2014

1 Introduction

The problem of tiling originated from antiquity, with the Ostomachion attributed to Archimedes. The basic premise of the problem is where a completed board is partitioned into smaller pieces. The specific problem of tiling a rectangular board with polyominoes is more recent, with earliest occurences in Japanese Tatami mats covering.

More recently, the advent of the computer has changed the way the problem has been tackled. The Dancing Links algorithm has been invented by Hiroshi Hitotsumatsu and Kohei Noshita in 1979 [1]. It has been used to tackle the ways to tile a finite alphabet of blocks into a rectangular grid. More generally, when the alphabet is not restricted to a finite set, a zero-suppressed binary decision diagram by Shinichi Minato [2] can be used to solve the problem.

However, these algorithms are both exponential in the size of the input. Hence, a fast algorithm is required if we wish to solve the problem efficiently.



2 Definitions

Definition 1. A **polyomino** is a plane figure consisting of squares connected edgewise. A n-omino is a polyomino with n squares. Refer to Figure 1 for examples of polyominoes.

Definition 2. **Alphabet** refers to a non-empty set of polyominoes which may be infinite.

Definition 3. A **board** is a polyomino which we attempt to partition into smaller polyominoes from a given alphabet.

Definition 4. A **tiling** of a board refers to a way to partition the board for some alphabet.

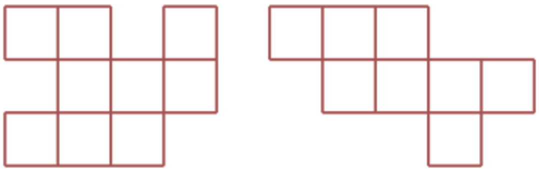


Figure 1: Examples of Polyominos: 9-omino (left); 8-omino (right)

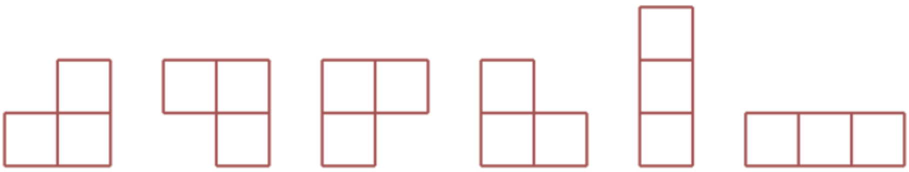


Figure 2: An alphabet of trominoes

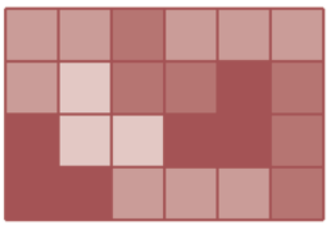


Figure 3: A tiling of a 4×6 board using trominoes

3 Known Results and Problem Statement

3.1 Known Results

In [5], the generating functions for tromino tiling of $2 \times 3k$ and $3 \times k$ boards are given:
For a $2 \times 3k$ board,

$$G_{T(2)}(x^3) = \frac{1 - x^3}{1 - 4x^3 + x^6}. \tag{1}$$

For a $3 \times k$ board,

$$G_{T(3)}(x) = \frac{x^3 - 1}{-1 + x + 3x^2 + 6x^3 + x^4 - x^6}. \tag{2}$$

3.2 Problem Statement

In this project, we aim to find the number of ways to tile a $4 \times 3k$ board given an alphabet consisting of only trominos.
The generating function we obtained is given by

$$\begin{aligned} G_{T(4)}(x) \;=\; & (-x^{14} + 45x^{13} - 790x^{12} + 7195x^{11} - 37791x^{10} + 120544x^9 - 241021x^8 + 307384x^7 \\ & -251359x^6 + 131039x^5 - 42817x^4 + 8472x^3 - 952x^2 + 53x - 1)/ \\ & (x^{15} - 56x^{14} + 1223x^{13} - 13643x^{12} + 87066x^{11} - 338409x^{10} + 836269x^9 - 1345297x^8 \\ & +1419177x^7 - 976456x^6 + 431092x^5 - 118633x^4 + 19424x^3 - 1761x^2 + 76x - 1) \end{aligned}$$

4 Methods

In order to properly explain the methodology, a few more definitions must be made.

Definition 5. A board is **right-aligned, horizontally-convex (RAHC)** if the following two conditions hold:

(a) All rows of squares either terminate on the same column or are empty.

(b) All rows of squares are continuous and have no gaps in between them.

An example of an RAHC board and a non-RAHC board is shown in Figure 4. A RAHC board can be expressed as a row vector $\{a_1, a_2, \ldots, a_n\}$ where a_i denotes the number of squares in row i. For example, the RAHC board in Figure 4 is given by $(4, 3, 5)$. Moreover, let $T(a_1, a_2, \ldots, a_n)$ denote the number of ways to tile $\{a_1, a_2, \ldots, a_n\}$. We define $T(a_1, a_2, \ldots, a_n) = 0$ if any of $a_1, a_2, \ldots, a_n$ are negative.

Definition 6. The **maximum point** of an RAHC board denotes the value p such that $a_p > a_i$ for all $i < p$ and $a_p \geq a_i$ for $i > p$. In other words, this denotes the top-most row with the most number of cells.

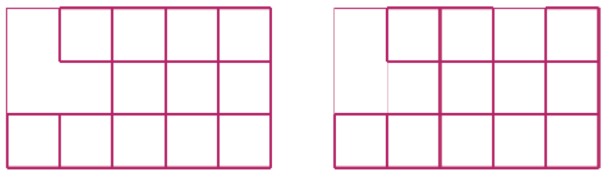


Figure 4: An example of an RAHC board and a non-RAHC board.

Definition 7. For an RAHC board with maximum point p, the left-most square on row p is the **top-left cell** of the RAHC board.

Definition 8. For an RAHC board given by the row vector $\{a_1, a_2, \ldots, a_n\}$ with maximum point p, the **effective value** of the RAHC board denotes the largest value e such that $a_p = a_{p-1} = \cdots = a_{p-e}$. In other words, the effective value denotes the number of consecutive rows after row p that have a_p squares in them.

4.1 Initial Methods

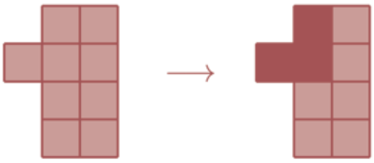
Originally, we attempted to extend the method used to find the generating function of tromino tiling $3 \times k$ rectangles as shown in [5]. However, finding the number of basic blocks of a $4 \times 3k$ rectangle is definitely not a simple task. We also tried various ways to colour the board hoping it would give us interesting properties regarding the tilings of such a board. However, these methods did not aid much in solving the problem. Hence, we looked into the idea of recursion.

4.2 Method X

Consider an RAHC board with row vector $\{a_1, \ldots, a_n\}$ and maximum point p. Note that in order to obtain a tiling of the RAHC board, a few trominoes can be chosen to be removed at each stage such that the top-left cell of the RAHC board is removed, while maintaining the property that the resulting board is RAHC. Hence, by considering the possible effective values (EV) of the board, a set of rules to remove the top-left cell of a RAHC board can be established.

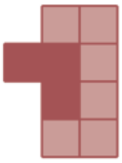
EV = 0

$$
\begin{aligned}
& T(a_1, a_2, \ldots, a_{p-1}, a_p, a_{p+1}, \ldots, a_n) \\
= \; & T(a_1, a_2, \ldots, a_{p-1} - 1, a_p - 2, a_{p+1}, \ldots, a_n) && \text{when } a_{p-1} = a_p - 1 \text{ and } a_p \geq 2 \\
& + T(a_1, a_2, \ldots, a_{p-1}, a_p - 2, a_{p+1} - 1, \ldots, a_n) && \text{when } a_{p+1} = a_p - 1 \text{ and } a_p \geq 2 \\
& + T(a_1, a_2, \ldots, a_{p-1}, a_p - 3, a_{p+1}, \ldots, a_n) && \text{when } a_p \geq 3.
\end{aligned}
$$

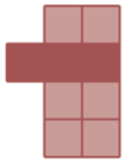


when $a_{p-1} = a_p - 1$, and $a_p \geq 2$;

and



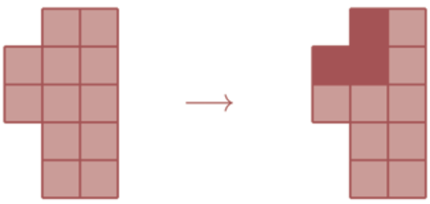
when $a_{p+1} = a_p - 1$, and $a_p \geq 2$; and



when $a_p \geq 3$

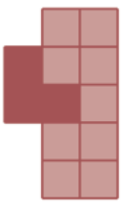
EV = 1

$$
\begin{aligned}
& T(a_1, a_2, \ldots, a_{p-1}, a_p, a_{p+1}, \ldots, a_n) \\
= \; & T(a_1, a_2, \ldots, a_{p-1} - 1, a_p - 2, a_{p+1}, \ldots, a_n) && \text{when } a_{p-1} = a_p - 1 \text{ and } a_p \geq 2 \\
& + T(a_1, a_2, \ldots, a_{p-1}, a_p - 1, a_{p+1} - 2, \ldots, a_n) && \text{when } a_p \geq 2 \\
& + T(a_1, a_2, \ldots, a_{p-1}, a_p - 2, a_{p+1} - 1, \ldots, a_n) && \text{when } a_p \geq 2 \\
& + T(a_1, a_2, \ldots, a_{p-1}, a_p - 3, a_{p+1}, \ldots, a_n) && \text{when } a_p \geq 3
\end{aligned}
$$

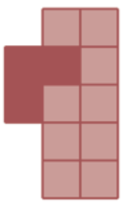


when $a_{p-1} = a_p - 1$, and $a_p \geq 2$;

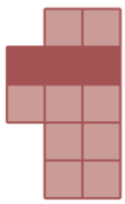
and



or



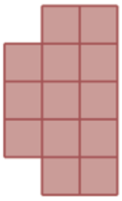
when $a_p \geq 2$; and



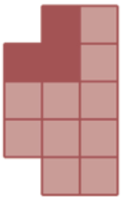
when $a_p \geq 3$

EV = 2

$$
\begin{aligned}
& T(a_1, a_2, \ldots, a_{p-1}, a_p, a_{p+1}, \ldots, a_n) & \\
= & T(a_1, a_2, \ldots, a_{p-1} - 1, a_p - 2, a_{p+1}, \ldots, a_n) & \text{when } a_{p-1} = a_p - 1 \text{ and } a_p \geq 2 \\
& + T(a_1, a_2, \ldots, a_{p-1}, a_p - 3, a_{p+1}, \ldots, a_n) & \text{when } a_p \geq 3 \\
& + T(a_1, a_2, \ldots, a_p - 1, a_{p+1} - 1, a_{p+2} - 1, \ldots, a_n) & \text{when } a_p \geq 1 \\
& + T(a_1, a_2, \ldots, a_p - 2, a_{p+1} - 2, a_{p+2} - 2, \ldots, a_n) & \text{when } a_p \geq 2 \\
& + T(a_1, a_2, \ldots, a_{p-1}, a_p - 2, a_{p+1} - 1, \ldots, a_n) & \text{when } a_p \geq 2 \\
& + T(a_1, a_2, \ldots, a_{p-1}, a_p - 1, a_{p+1} - 2, \ldots, a_n) & \text{when } a_p \geq 2
\end{aligned}
$$

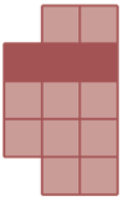


$\longrightarrow$

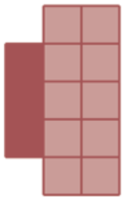


when $a_{p-1} = a_p - 1$, and $a_p \geq 2$;

and

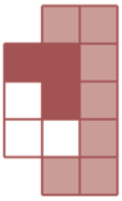


when $a_p \geq 3$; and

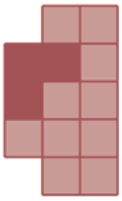


when $a_p \geq 1$;

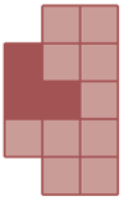
and



or



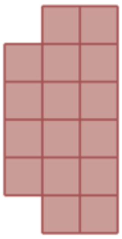
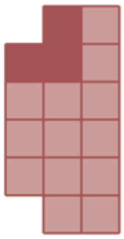
or

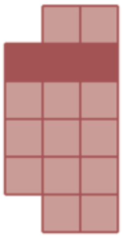
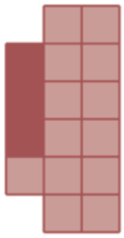


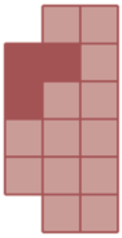
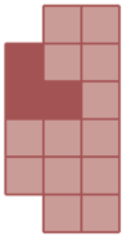
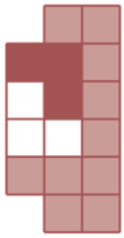
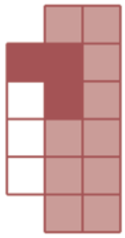
when $a_p \geq 2$

EV ≥ 3

$$
\begin{aligned}
& T(a_1, a_2, \ldots, a_{p-1}, a_p, a_{p+1}, \ldots, a_n) & \\
= & T(a_1, a_2, \ldots, a_{p-1} - 1, a_p - 2, a_{p+1}, \ldots, a_n) & \text{when } a_{p-1} = a_p - 1 \text{ and } a_p \geq 2 \\
& + T(a_1, a_2, \ldots, a_{p-1}, a_p - 3, a_{p+1}, \ldots, a_n) & \text{when } a_p \geq 3 \\
& + T(a_1, a_2, \ldots, a_p - 1, a_{p+1} - 1, a_{p+2} - 1, \ldots, a_n) & \text{when } a_p \geq 1 \\
& + T(a_1, a_2, \ldots, a_{p-1}, a_p - 2, a_{p+1} - 1, \ldots, a_n) & \text{when } a_p \geq 2 \\
& + T(a_1, a_2, \ldots, a_{p-1}, a_p - 1, a_{p+1} - 2, \ldots, a_n) & \text{when } a_p \geq 2 \\
& + T(a_1, a_2, \ldots, a_p - 2, a_{p+1} - 2, a_{p+2} - 2, \ldots, a_n) & \text{when } a_p \geq 2 \\
& + T(a_1, a_2, \ldots, a_p - 2, a_{p+1} - 2, a_{p+2} - 1, a_{p+3} - 1, \ldots, a_n) & \text{when } a_p \geq 2
\end{aligned}
$$

 $\longrightarrow$  when $a_{p-1} = a_p - 1,$ and $a_p \geq 2$

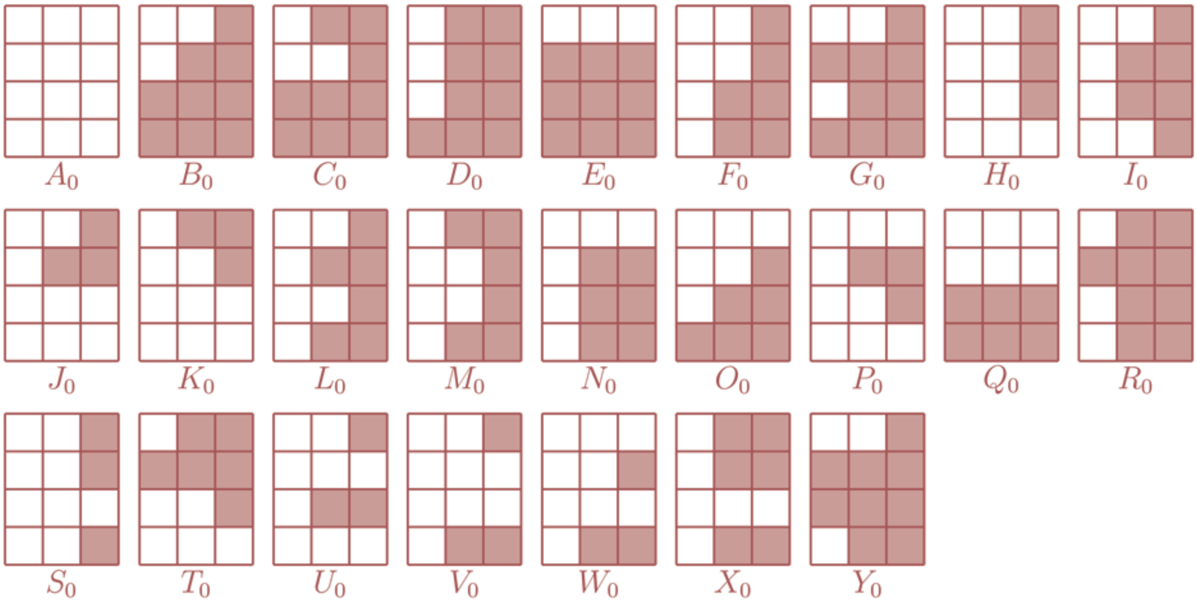
and  when $a_p \geq 3$; and  when $a_p \geq 1$;

and  or  or  or  when $a_p \geq 2$

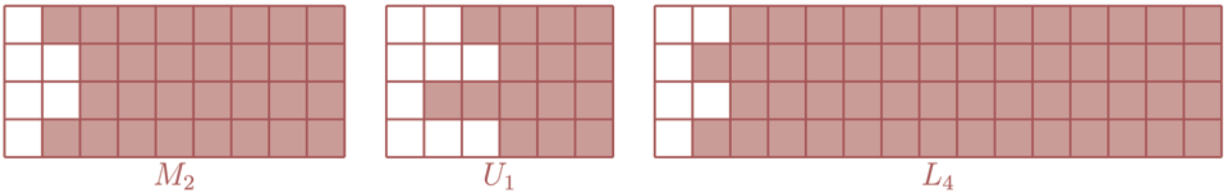
This algorithm was then implemented on a computer program to aid in finding the number of ways to tile a $4\times 3n$ board for small values of n. These results will be shown in the **Results** section of this report.

4.3 Method M

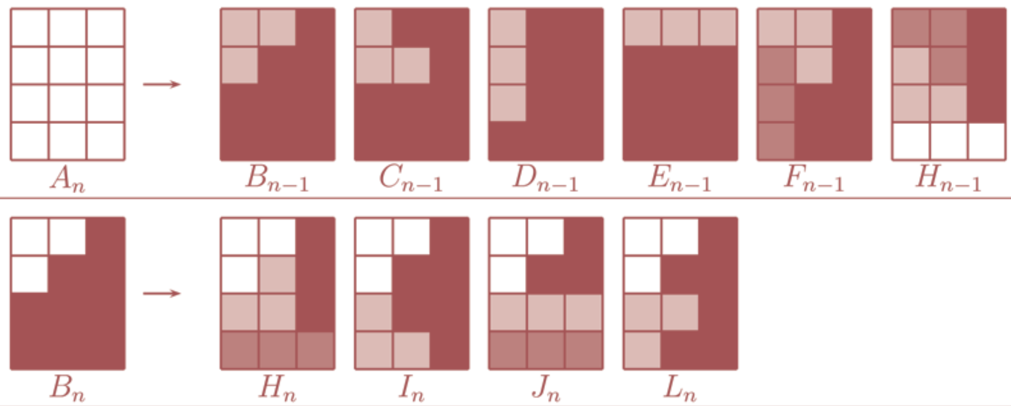
Method M uses the rules of Method X to partition an RAHC board, but in this method we aim to come up with a recursive relationship by considering all possible states of an RAHC Board. First, we define 25 states as follows:

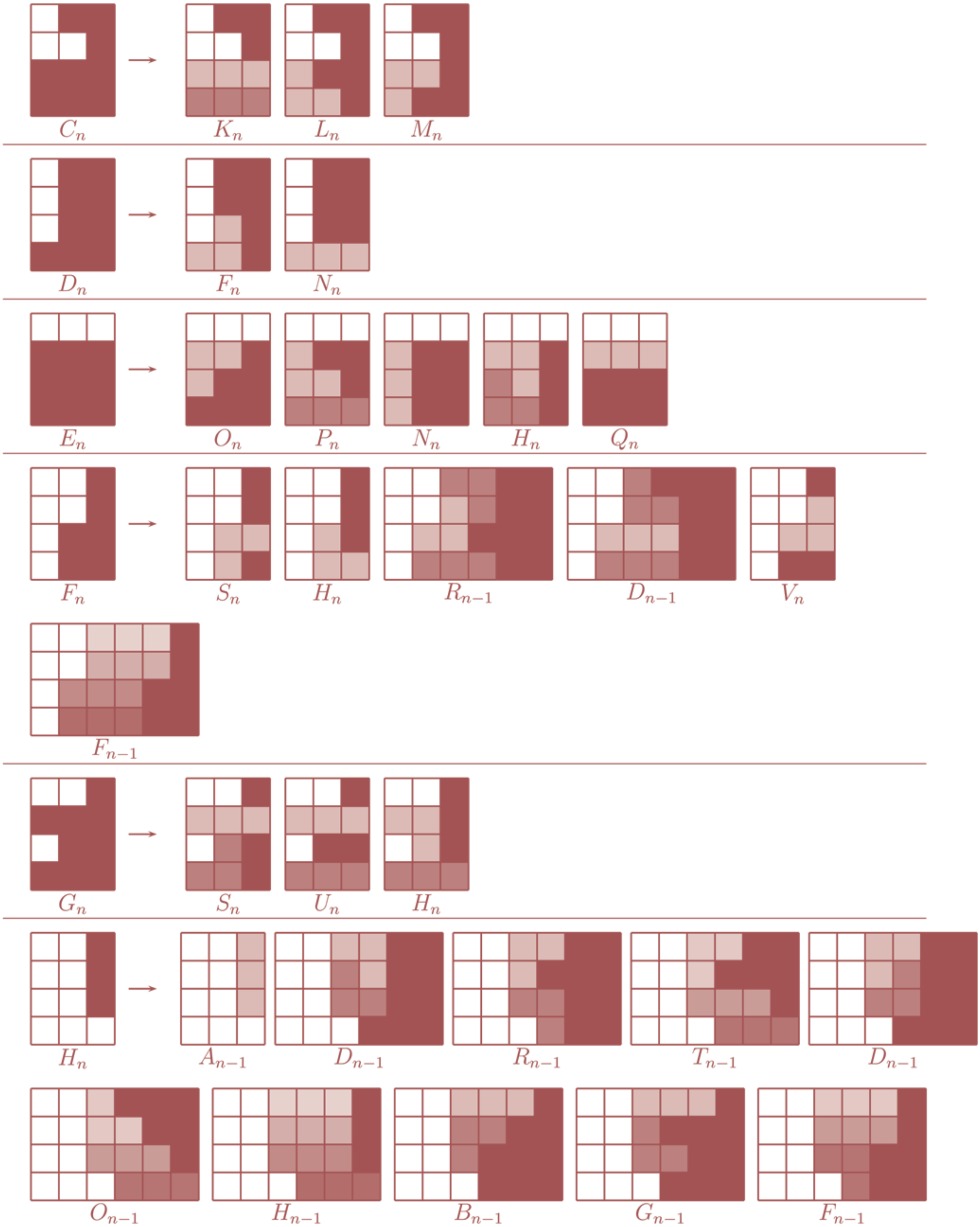


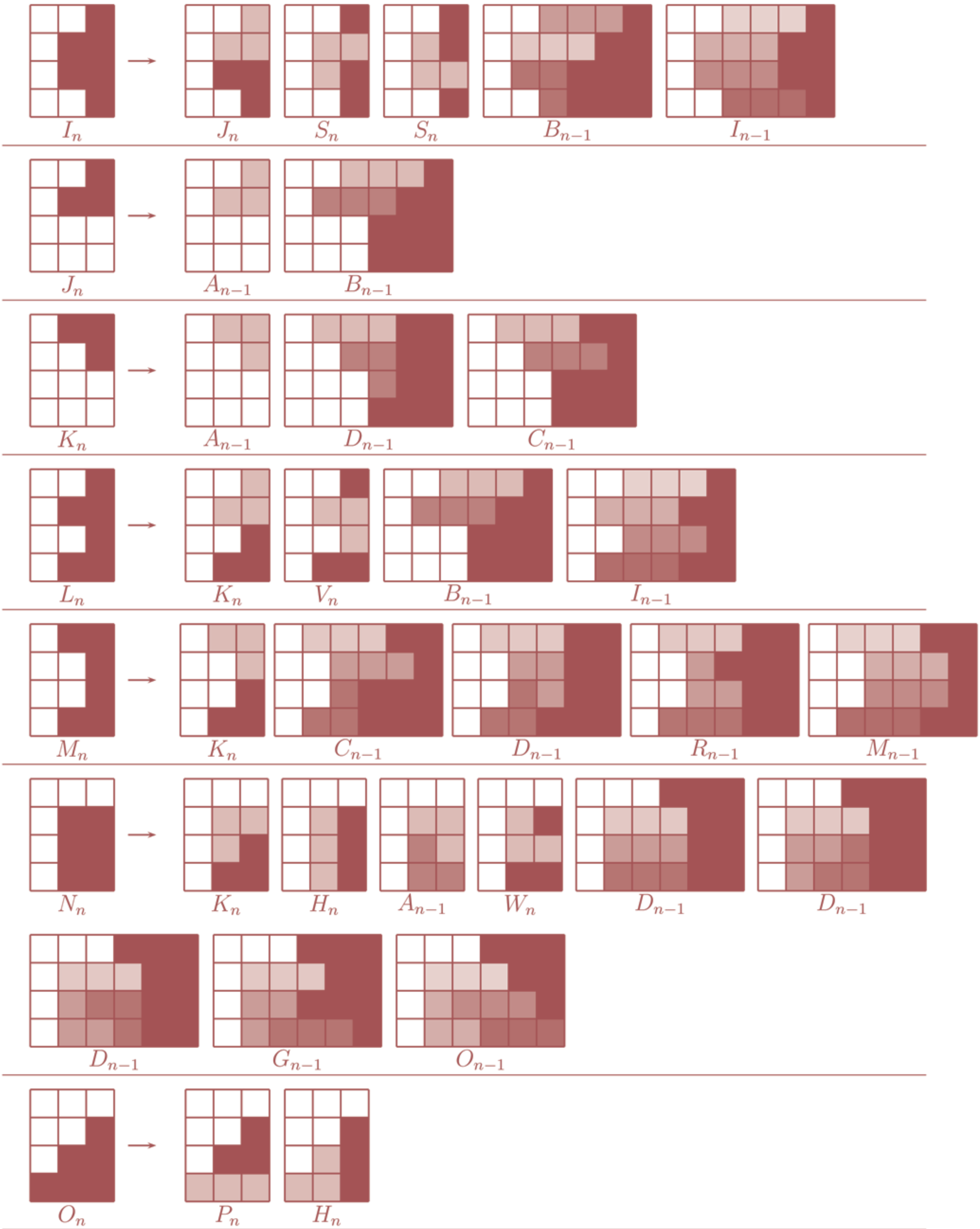
We define A_n (resp. B_n to Y_n) as A_0 (resp. B_0 to Y_0) with a $4 \times 3n$ rectangular board to its right. Examples of some boards are shown below.

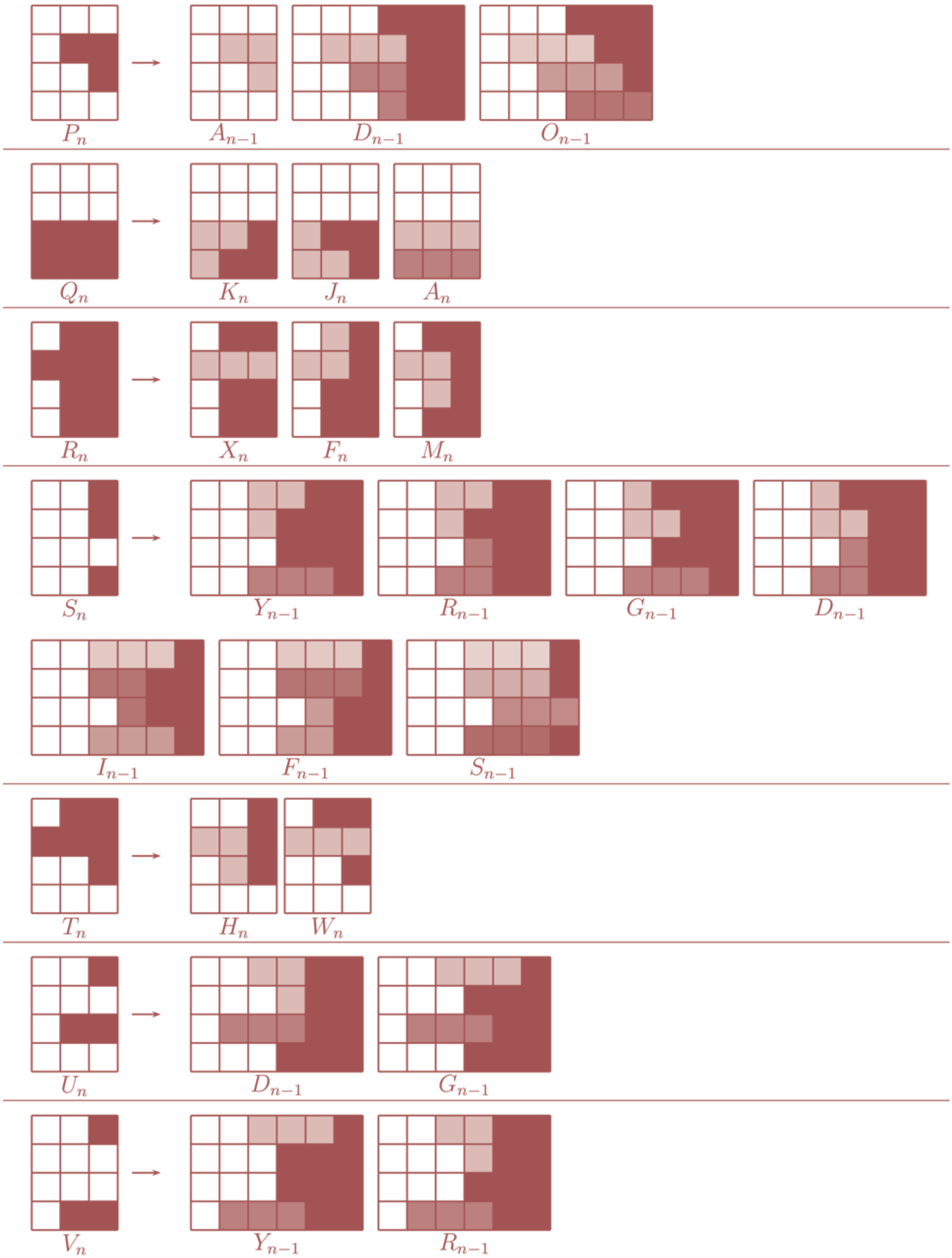


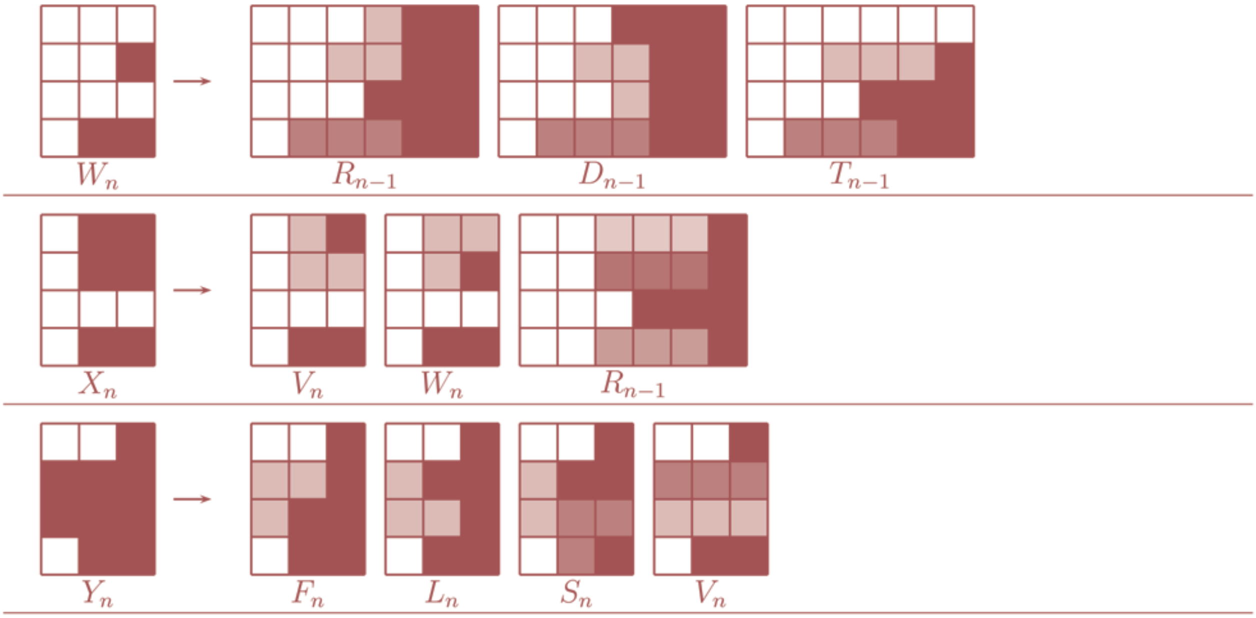
Now, note that each state has a specific group of states that they can move on to. The diagrams below show all possible states that each state can reach, and how one can tile the RAHC board at one state to reach another state.




C_n → K_n L_n M_n
D_n → F_n N_n
E_n → O_n P_n N_n H_n Q_n
F_n → S_n H_n R_{n-1} D_{n-1} V_n F_{n-1}
G_n → S_n U_n H_n
H_n → A_{n-1} D_{n-1} R_{n-1} T_{n-1} D_{n-1} O_{n-1} H_{n-1} B_{n-1} G_{n-1} F_{n-1}


I_n → J_n S_n S_n B_{n-1} I_{n-1}
J_n → A_{n-1} B_{n-1}
K_n → A_{n-1} D_{n-1} C_{n-1}
L_n → K_n V_n B_{n-1} I_{n-1}
M_n → K_n C_{n-1} D_{n-1} R_{n-1} M_{n-1}
N_n → K_n H_n A_{n-1} W_n D_{n-1} D_{n-1} D_{n-1} G_{n-1} O_{n-1}
O_n → P_n H_n


P_n A_{n-1} D_{n-1} O_{n-1}
Q_n K_n J_n A_n
R_n X_n F_n M_n
S_n Y_{n-1} R_{n-1} G_{n-1} D_{n-1}
I_{n-1} F_{n-1} S_{n-1}
T_n H_n W_n
U_n D_{n-1} G_{n-1}
V_n Y_{n-1} R_{n-1}



Therefore, based on this, we will be able to obtain a 25 degree recurrence relationship as shown:

$$A_n = B_{n-1} + C_{n-1} + D_{n-1} + E_{n-1} + F_{n-1} + H_{n-1}$$
$$B_n = H_n + I_n + J_n + L_n$$
$$C_n = K_n + L_n + M_n$$
$$D_n = F_n + N_n$$
$$E_n = H_n + N_n + O_n + P_n + Q_n$$
$$F_n = D_{n-1} + F_{n-1} + H_n + R_{n-1} + S_n + V_n$$
$$G_n = H_n + S_n + U_n$$
$$H_n = A_{n-1} + B_{n-1} + 2D_{n-1} + F_{n-1} + G_{n-1} + H_{n-1} + O_{n-1} + R_{n-1} + T_{n-1}$$
$$I_n = B_{n-1} + I_{n-1} + J_n + 2S_n$$
$$J_n = A_{n-1} + B_{n-1}$$
$$K_n = A_{n-1} + C_{n-1} + D_{n-1}$$
$$L_n = B_{n-1} + K_n + L_{n-1} + V_n$$
$$M_n = C_{n-1} + D_{n-1} + K_n + M_{n-1} + R_{n-1}$$
$$N_n = A_{n-1} + 3D_{n-1} + G_{n-1} + H_n + K_n + O_{n-1} + W_n$$
$$O_n = H_n + P_n$$
$$P_n = A_{n-1} + D_{n-1} + O_{n-1}$$

$$Q_n = A_n + J_n + K_n$$
$$R_n = F_n + M_n + X_n$$
$$S_n = D_{n-1} + F_{n-1} + G_{n-1} + I_{n-1} + R_{n-1} + S_{n-1} + Y_{n-1}$$
$$T_n = H_n + W_n$$
$$U_n = D_{n-1} + G_{n-1}$$
$$V_n = R_{n-1} + Y_{n-1}$$
$$W_n = D_{n-1} + R_{n-1} + T_{n-1}$$
$$X_n = R_{n-1} + V_n + W_n$$
$$Y_n = F_n + L_n + S_n + V_n$$

5 Results

After implementing the algorithm described in Method X on a computer program, we obtain the following values. Only the first few values are obtained due to integer overflow.

n	Number of ways to tile a $4 \times 3n$ rectangle
0	1
1	23
2	939
3	41813
4	1895145
5	86208957
6	3924499731
7	178682349823
8	8135650498647

Now let us look at the equations obtained in Method M. Note that it is possible to eliminate some of the functions easily, for example, consider the function B_n. It can be substituted with $H_n + I_n + J_n + L_n$ in the following equations and eliminated from the original 25 degree recurrence relation:

$$A_n = (H_{n-1} + I_{n-1} + J_{n-1} + L_{n-1}) + C_{n-1} + D_{n-1} + E_{n-1} + F_{n-1} + H_{n-1}$$
$$H_n = A_{n-1} + (H_{n-1} + I_{n-1} + J_{n-1} + L_{n-1}) + 2D_{n-1} + F_{n-1} + G_{n-1} + H_{n-1} + O_{n-1} + R_{n-1} + T_{n-1}$$
$$I_n = (H_{n-1} + I_{n-1} + J_{n-1} + L_{n-1}) + I_{n-1} + J_n + 2S_n$$
$$J_n = A_{n-1} + (H_{n-1} + I_{n-1} + J_{n-1} + L_{n-1})$$
$$L_n = (H_{n-1} + I_{n-1} + J_{n-1} + L_{n-1}) + K_n + L_{n-1} + V_n$$

Similarly the function C_n can be substituted with $K_n + L_n + M_n$ in the following equations and eliminated from the recurrence relation:

$$A_n = (H_{n-1} + I_{n-1} + J_{n-1} + L_{n-1}) + (K_{n-1} + L_{n-1} + M_{n-1}) + D_{n-1} + E_{n-1} + F_{n-1} + H_{n-1}$$
$$K_n = A_{n-1} + (K_{n-1} + L_{n-1} + M_{n-1}) + D_{n-1}$$
$$M_n = (K_{n-1} + L_{n-1} + M_{n-1}) + D_{n-1} + K_n + M_{n-1} + R_{n-1}$$

In the same manner, we can eliminate the functions $D_n, E_n, G_n, O_n, Q_n, R_n, T_n, Y_n$. Consequently, we obtain a 15 degree recurrence relation:

$$
\begin{aligned}
A_n &= A_{n-1} + 2F_{n-1} + 4H_{n-1} + I_{n-1} + 2J_{n-1} + 2K_{n-1} + 2L_{n-1} + M_{n-1} + 2N_{n-1} + 2P_{n-1} \\
F_n &= A_{n-1} + 16F_{n-1} + 10H_{n-1} + 3I_{n-1} + 3J_{n-1} + 2K_{n-1} + 5L_{n-1} + 5M_{n-1} + 7N_{n-1} \\
&\quad + 3P_{n-1} + 5S_{n-1} + 2U_{n-1} + 2V_{n-1} + W_{n-1} + 4X_{n-1} \\
H_n &= A_{n-1} + 7F_{n-1} + 9H_{n-1} + 2I_{n-1} + 3J_{n-1} + 2K_{n-1} + 3L_{n-1} + 2M_{n-1} + 5N_{n-1} + 3P_{n-1} \\
&\quad + S_{n-1} + U_{n-1} + W_{n-1} + X_{n-1} \\
I_n &= A_{n-1} + 10F_{n-1} + 8H_{n-1} + 6I_{n-1} + 4J_{n-1} + 2K_{n-1} + 6L_{n-1} + 3M_{n-1} + 4N_{n-1} + 2P_{n-1} \\
&\quad + 6S_{n-1} + 2U_{n-1} + 2V_{n-1} + 2X_{n-1} \\
J_n &= A_{n-1} + 2F_{n-1} + 5H_{n-1} + 2I_{n-1} + 3J_{n-1} + 2K_{n-1} + 3L_{n-1} + M_{n-1} + 2N_{n-1} + 2P_{n-1} \\
K_n &= A_{n-1} + 3F_{n-1} + 4H_{n-1} + I_{n-1} + 2J_{n-1} + 3K_{n-1} + 3L_{n-1} + 2M_{n-1} + 3N_{n-1} + 2P_{n-1} \\
L_n &= A_{n-1} + 5F_{n-1} + 5H_{n-1} + 2I_{n-1} + 3J_{n-1} + 3K_{n-1} + 6L_{n-1} + 3M_{n-1} + 3N_{n-1} + 2P_{n-1} \\
&\quad + S_{n-1} + V_{n-1} + X_{n-1} \\
M_n &= A_{n-1} + 6F_{n-1} + 4H_{n-1} + I_{n-1} + 2J_{n-1} + 4K_{n-1} + 4L_{n-1} + 6M_{n-1} + 4N_{n-1} + 2P_{n-1} + 2X_{n-1} \\
N_n &= 3A_{n-1} + 17F_{n-1} + 20H_{n-1} + 4I_{n-1} + 7J_{n-1} + 8L_{n-1} + 6M_{n-1} + 14N_{n-1} \\
&\quad + 8P_{n-1} + 2S_{n-1} + 2U_{n-1} + 2W_{n-1} + 2X_{n-1} \\
P_n &= A_{n-1} + 3F_{n-1} + 5H_{n-1} + I_{n-1} + 2J_{n-1} + 2K_{n-1} + 2L_{n-1} + M_{n-1} + 3N_{n-1} + 3P_{n-1} \\
S_n &= 4F_{n-1} + H_{n-1} + I_{n-1} + L_{n-1} + M_{n-1} + N_{n-1} + 3S_{n-1} + U_{n-1} + V_{n-1} + X_{n-1} \\
U_n &= F_{n-1} + H_{n-1} + N_{n-1} + S_{n-1} + U_{n-1} \\
V_n &= 2F_{n-1} + L_{n-1} + M_{n-1} + S_{n-1} + V_{n-1} + X_{n-1} \\
W_n &= 2F_{n-1} + H_{n-1} + M_{n-1} + N_{n-1} + W_{n-1} + X_{n-1} \\
X_n &= 5F_{n-1} + H_{n-1} + L_{n-1} + 3M_{n-1} + N_{n-1} + S_{n-1} + V_{n-1} + W_{n-1} + 3X_{n-1}.
\end{aligned}
$$

Note that we can express the above system of linear equations in a matrix multiplication as shown:

$$\begin{pmatrix} A_n \\ F_n \\ H_n \\ I_n \\ J_n \\ K_n \\ L_n \\ M_n \\ N_n \\ P_n \\ S_n \\ U_n \\ V_n \\ W_n \\ X_n \end{pmatrix} = \begin{pmatrix} 1 & 2 & 4 & 1 & 2 & 2 & 2 & 1 & 2 & 2 & 0 & 0 & 0 & 0 & 0 \\ 1 & 16 & 10 & 3 & 3 & 2 & 5 & 5 & 7 & 3 & 5 & 2 & 2 & 1 & 4 \\ 1 & 7 & 9 & 2 & 3 & 2 & 3 & 2 & 5 & 3 & 1 & 1 & 0 & 1 & 1 \\ 1 & 10 & 8 & 6 & 4 & 2 & 6 & 3 & 4 & 2 & 6 & 2 & 2 & 0 & 2 \\ 1 & 2 & 5 & 2 & 3 & 2 & 3 & 1 & 2 & 2 & 0 & 0 & 0 & 0 & 0 \\ 1 & 3 & 4 & 1 & 2 & 3 & 3 & 2 & 3 & 2 & 0 & 0 & 0 & 0 & 0 \\ 1 & 5 & 5 & 2 & 3 & 3 & 6 & 3 & 3 & 2 & 1 & 0 & 1 & 0 & 1 \\ 1 & 6 & 4 & 1 & 2 & 4 & 4 & 6 & 4 & 2 & 0 & 0 & 0 & 0 & 2 \\ 3 & 17 & 20 & 4 & 7 & 7 & 8 & 6 & 14 & 8 & 2 & 2 & 0 & 2 & 2 \\ 1 & 3 & 5 & 1 & 2 & 2 & 2 & 1 & 3 & 3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 4 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 3 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 2 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 5 & 1 & 0 & 0 & 0 & 1 & 3 & 1 & 0 & 1 & 0 & 1 & 1 & 3 \end{pmatrix} \begin{pmatrix} A_{n-1} \\ F_{n-1} \\ H_{n-1} \\ I_{n-1} \\ J_{n-1} \\ K_{n-1} \\ L_{n-1} \\ M_{n-1} \\ N_{n-1} \\ P_{n-1} \\ S_{n-1} \\ U_{n-1} \\ V_{n-1} \\ W_{n-1} \\ X_{n-1} \end{pmatrix}$$

From Method X, we are able to obtain that $A_0 = F_0 = H_0 = J_0 = K_0 = L_0 = M_0 = P_0 = 1$, $N_0 = 3$ and $S_0 = U_0 = V_0 = W_0 = X_0 = 0$. Hence, we can express the above expression as follows:

$$\begin{pmatrix} A_n \\ F_n \\ H_n \\ I_n \\ J_n \\ K_n \\ L_n \\ M_n \\ N_n \\ P_n \\ S_n \\ U_n \\ V_n \\ W_n \\ X_n \end{pmatrix} = \begin{pmatrix} 1 & 2 & 4 & 1 & 2 & 2 & 2 & 1 & 2 & 2 & 0 & 0 & 0 & 0 & 0 \\ 1 & 16 & 10 & 3 & 3 & 2 & 5 & 5 & 7 & 3 & 5 & 2 & 2 & 1 & 4 \\ 1 & 7 & 9 & 2 & 3 & 2 & 3 & 2 & 5 & 3 & 1 & 1 & 0 & 1 & 1 \\ 1 & 10 & 8 & 6 & 4 & 2 & 6 & 3 & 4 & 2 & 6 & 2 & 2 & 0 & 2 \\ 1 & 2 & 5 & 2 & 3 & 2 & 3 & 1 & 2 & 2 & 0 & 0 & 0 & 0 & 0 \\ 1 & 3 & 4 & 1 & 2 & 3 & 3 & 2 & 3 & 2 & 0 & 0 & 0 & 0 & 0 \\ 1 & 5 & 5 & 2 & 3 & 3 & 6 & 3 & 3 & 2 & 1 & 0 & 1 & 0 & 1 \\ 1 & 6 & 4 & 1 & 2 & 4 & 4 & 6 & 4 & 2 & 0 & 0 & 0 & 0 & 2 \\ 3 & 17 & 20 & 4 & 7 & 7 & 8 & 6 & 14 & 8 & 2 & 2 & 0 & 2 & 2 \\ 1 & 3 & 5 & 1 & 2 & 2 & 2 & 1 & 3 & 3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 4 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 3 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 2 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 5 & 1 & 0 & 0 & 0 & 1 & 3 & 1 & 0 & 1 & 0 & 1 & 1 & 3 \end{pmatrix}^n \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 3 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Hence, using this matrix, we will be able to calculate the value of A_n, and this gives us the number of ways to tile a $4 \times 3n$.

From the system of recurrence relation, we can also get the generating function given in section 3.2:

$$\begin{aligned} T(x) \;=\; & (-x^{14} + 45x^{13} - 790x^{12} + 7195x^{11} - 37791x^{10} + 120544x^{9} - 241021x^{8} + 307384x^{7} \\ & -251359x^{6} + 131039x^{5} - 42817x^{4} + 8472x^{3} - 952x^{2} + 53x - 1)/ \\ & (x^{15} - 56x^{14} + 1223x^{13} - 13643x^{12} + 87066x^{11} - 338409x^{10} + 836269x^{9} - 1345297x^{8} \\ & +1419177x^{7} - 976456x^{6} + 431092x^{5} - 118633x^{4} + 19424x^{3} - 1761x^{2} + 76x - 1) \end{aligned}$$

References

[1] Hitotumatu, Hirosi and Noshita, Kohei (1979), *A Technique for Implementing Backtrack Algorithms and its Application*. Information Processing Letters 8 (4): 174–175, doi:10.1016/0020-0190(79)90016-4

[2] Shin-ichi Minato (1993), *Zero-suppressed BDDs for set manipulation in combinatorial problems*, DAC '93: Proceedings of the 30th international conference on Design automation.

[3] Erica Klarreich (1991), *Counting Domino Tilings of Mutilated Grids*, OSU REU '91.

[4] Bodini, Olivier and Latapy, Matthieu (2003), *Generalized Tilings with Height Functions*, Morsmos 7 (1): 4768, ISSN 1870-6525.

[5] Silvia Heubach, Phyllis Chinn, Patrick Callahan. *Tiling With Trominoes*

http://web.calstatela.edu/faculty/sheubac/papers/triomino tilings.pdf

The authors are students of NUS High School of Mathematics and Sciences.