Hex-Chaos Compositions and Equivalence Classes of Packing Problems

Gary R. Greenfield

University of Richmond, Virginia, USA; ggreenfi@richmond.edu

Abstract

We consider a generative art scheme that uses zero-one labelings of the edges of all the cells of a grid and chaotic one-dimensional cellular automata to assign hexadecimal digits to the cells of the grid. This allows us to color all the cells by mapping hex digits to colors. We invoke a genetic algorithm to maximize the number of occurrences of two hex digits thereby evolving what we call hex-chaos compositions. Using elementary group theory we prove, up to equivalence, there are thirteen different types of hex-chaos compositions.

Introduction

Let $\mathcal{G}_{m,n}$ denote the the set of $m \times n$ grids whose cells are labeled with hexadecimal digits. If $\mathcal{X} = \{0, \ldots, 9, a, \ldots, f\}$, then $\mathcal{G}_{m,n} = \{\boxed{x_{r,c}} : x_{r,c} \in \mathcal{X},$ for $1 \leq r \leq m, 1 \leq c \leq n\}$. Each cell of a hex grid $X = \boxed{x_{r,c}} \in \mathcal{G}_{m,n}$ has four edges. Suppose the edges of all the cells are labelled using zeros and ones. Then a generic cell $x_{r,c}$ has top, left, bottom and right edge labels $h_{r,c}$, $v_{r,c}$, $h_{r+1,c}$ and $v_{r,c+1}$. The horizontal edge labels determine an $(m+1) \times n$ matrix $H = (h_{p,q})$, while the vertical edge labels determine an $m \times (n+1)$ matrix $V = (v_{s,t})$. We say X is a *pullback* if there exist H and V such that for all r and c, $x_{r,c}$ is equal to the conversion of the base two representation of the four digit binary number formed by concatenating the edge labels counterclockwise from the top to a hex digit *i.e.,* $x_{r,c}$ equals the hex value of the decimal

$$8h_{r,c} + 4v_{r,c} + 2h_{r+1,c} + v_{r,c+1}.$$

In this case, we write $X = H \diamond V$. We denote the set of $m \times n$ hex grids that are pullbacks by $\mathcal{P}_{m,n}$. Evidently, the way to construct pullbacks is to interleave matrices of zeros and ones of the appropriate dimensions and then use the binary to hex conversion formula to determine the hex labels.

Example 1. Let $m = 4$, $n = 5$. Set

$$H = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{pmatrix}, \quad V = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}.$$

Then interleaving gives

$$X = \begin{bmatrix} & 1 & & 1 & & 0 & & 0 & & 1 & \\ 1 & x_{1,1} & 1 & x_{1,2} & 0 & x_{1,3} & 1 & x_{1,4} & 1 & x_{1,5} & 0 \\ & 1 & & 1 & & 1 & & 0 & & 0 & \\ 1 & x_{2,1} & 0 & x_{2,2} & 1 & x_{2,3} & 0 & x_{2,4} & 1 & x_{2,5} & 1 \\ & 1 & & 1 & & 1 & & 0 & & 0 & \\ 0 & x_{3,1} & 0 & x_{3,2} & 1 & x_{3,3} & 0 & x_{3,4} & 0 & x_{3,5} & 1 \\ & 1 & & 1 & & 0 & & 0 & & 1 & \\ 0 & x_{4,1} & 1 & x_{4,2} & 1 & x_{4,3} & 0 & x_{4,4} & 0 & x_{4,5} & 1 \\ & 0 & & 1 & & 1 & & 1 & & 0 & \end{bmatrix}.$$

and using the binary to hex conversion formula gives the pullback

$$X = H \diamond V = \begin{array}{|c|c|c|c|c|} \hline f & e & 3 & 5 & c \\ \hline e & b & e & 1 & 5 \\ \hline a & b & c & 0 & 3 \\ \hline 9 & f & 6 & 2 & 9 \\ \hline \end{array}.$$

For visualization purposes we assign a color ramp of sixteen shades of blue to the hex values 0 through f. To add visual interest to the construction we choose two hex values $0 \leq x_1 < x_2 \leq f$ and recolor them so they they stand out. If we choose $x_1 = 5$ and $x_2 = 9$ and assign them the colors brown and yellow respectively, we obtain the visualization for the X of Example 1 shown in Figure 1.

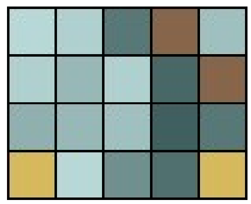

Figure 1: *A visualization of the hex labeled grid in Example 1 obtained by mapping* 5 *to brown,* 9 *to yellow and all other hex labels to shades of blue.*

Adding Chaos

Following an idea first proposed by Cruz *et al.* [1], we now require the edge labels of our pullbacks to be the outputs of iterates of chaotic or eventually periodic one-dimensional cellular automata [3]. Thus, we let H_0 be an $(m + 1) \times n$ matrix with randomly generated zeros and ones and V_0 be an $m \times (n + 1)$ matrix of randomly generated zeros and ones. We assign one dimensional automata to the $m + 1$ rows of H_0 and the $n + 1$ columns of V_0. We define H_i and V_i inductively by letting H_{i+1} and V_{i+1} be the matrices obtained by treating the rows of the H_i matrix and the columns of the V_i matrix as the inputs to the respective automata, yielding the rows of the H_{i+1} matrix and the columns of the V_{i+1} matrix as the outputs.

Example 2. Let $m = 4$, $n = 5$. Choose Rule 30 for the rows and Rule 54 for the columns. Let

$$H_0 = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 \end{pmatrix}, \quad V_0 = \begin{pmatrix} 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}.$$

Then H_{500} and V_{500} are the matrices H and V that gave rise to the pullback grid X of Example 1.

Hex-Chaos Compositions

Let $m = 30$, $n = 40$, and continue to use Rule 30 for rows and Rule 54 for columns. Towards the Cruz *et al.* goal of “harnessing chaos” we consider the following scheme for maximizing the number of occurrences of the two distinguished hex values x_1 and x_2 in the central region of a pullback hex grid. If X_{500} is a pullback induced after 500 iterations starting from H_0 and V_0 (i.e., $X_{500} = H_{500} \diamond V_{500}$), let c_i be the number of occurrences of x_i in X_{500}, w_i be the number of occurrences of x_i in the centered $3/5m \times 3/5n$ subgrid of X_{500} and $o_i = c_i - w_i$. Then the fitness of X_{500} is defined to be

$$F(X_{500}) = \min(c_1, c_2) + (w_1 + w_2) - (o_1 + o_2).$$

This function preferences pullbacks where the number of x_1's and x_2's are nearly equal, rewards pullbacks where occurrences of x_1 and x_2 are within the central window and penalizes pullbacks where occurrences of x_1 and x_2 are outside the central window.

We invoke a so-called 1 + 1 genetic algorithm by forming H_0' and V_0' from H_0 and V_0 by changing either one entry of H_0, one entry of V_0 or one entry of both, and replacing X_0 by X_0' whenever $F(X_{500}') \geq F(X_{500})$. This implements simple hill climbing such that the rows and columns of H_0 and V_0 are, via indirect feedback, forced to cooperate to maximize fitness. We let the algorithm run for 25,000 generations. That is, we make 25,000 attempts to improve fitness. Further details about the motivation, algorithm design and choice of parameter settings can be found in Greenfield [2].

Figure 2 shows three examples of hex-chaos compositions evolved using our evolutionary algorithm. From a distance they may look quite similar, but up close each one presents a different visual challenge to the viewer. Namely, what are the *rules* for the two distinguished colors? For example, for browns and yellows one finds: browns can be vertically or horizontally adjacent to each other, yellows cannot, and if brown and yellow are ever horizontally (respectively, vertically) adjacent then yellow must be on the left (respectively, on top). We leave it to the reader to infer the adjacency rules for the other two examples.

Packing Problems

Since we are trying to pack approximately an equal number of x_1's and x_2's into the central window, this raises the question of how many occurrences of x_1 and x_2 one can pack into an $m \times n$ pullback in general. Of course to consider such a question we will have to drop the assumption that the pullback is an iterate. Thus we wish to consider the problem of determining

$$\Gamma_{m,n}(x_1, x_2) = \max_{P \in \mathcal{P}_{m,n}} \min(\eta_{x_1}(P), \eta_{x_2}(P)),$$

where, for $x \in \mathcal{X}$ and $G \in \mathcal{G}$, $\eta_x(G)$ equals the number of occurrences of x in G. This is a challenging problem. We offer the following examples whose proofs appear in Greenfield [2].

Theorem 3. Let $x_1 = 5$, $x_2 = f$. Then $\Gamma_{m,n}(5, f)$ equals $\lfloor mn/2 \rfloor - 1$ if m is even and n is odd, and $\lfloor mn/2 \rfloor$ otherwise.

Theorem 4. Let $x_1 = 0$, $x_2 = f$. Then

$$\Gamma_{m,n}(1,2) = \begin{cases} \max((m-1)n/2, m(n-1)/2) & \text{if } m, n \text{ are odd} \\ \max((m-2)n/2 + n/2, m(n-2)/2 + m/2) & \text{if } m, n \text{ are even} \\ \max((m-2)n/2 + (n-1)/2, m(n-1)/2) & \text{if } m \text{ is even, } n \text{ is odd} \\ \max((m-1)n/2, m(n-2)/2 + (m-1)/2) & \text{if } m \text{ is odd, } n \text{ is even.} \end{cases}$$

The proofs construct P's for which $\eta_{x_1}(P) = \eta_{x_2}(P)$. If we let

$$\mathcal{E}_{m,n}(x_1, x_2) = \{P \in \mathcal{P}_{m,n} : \eta_{x_1}(P) = \eta_{x_2}(P)\},$$

and define

$$\Upsilon_{m,n}(x_1, x_2) = max_{E \in \mathcal{E}_{m,n}(x_1,x_2)}\, \eta_{x_1}(E),$$

this leads to the conjecture

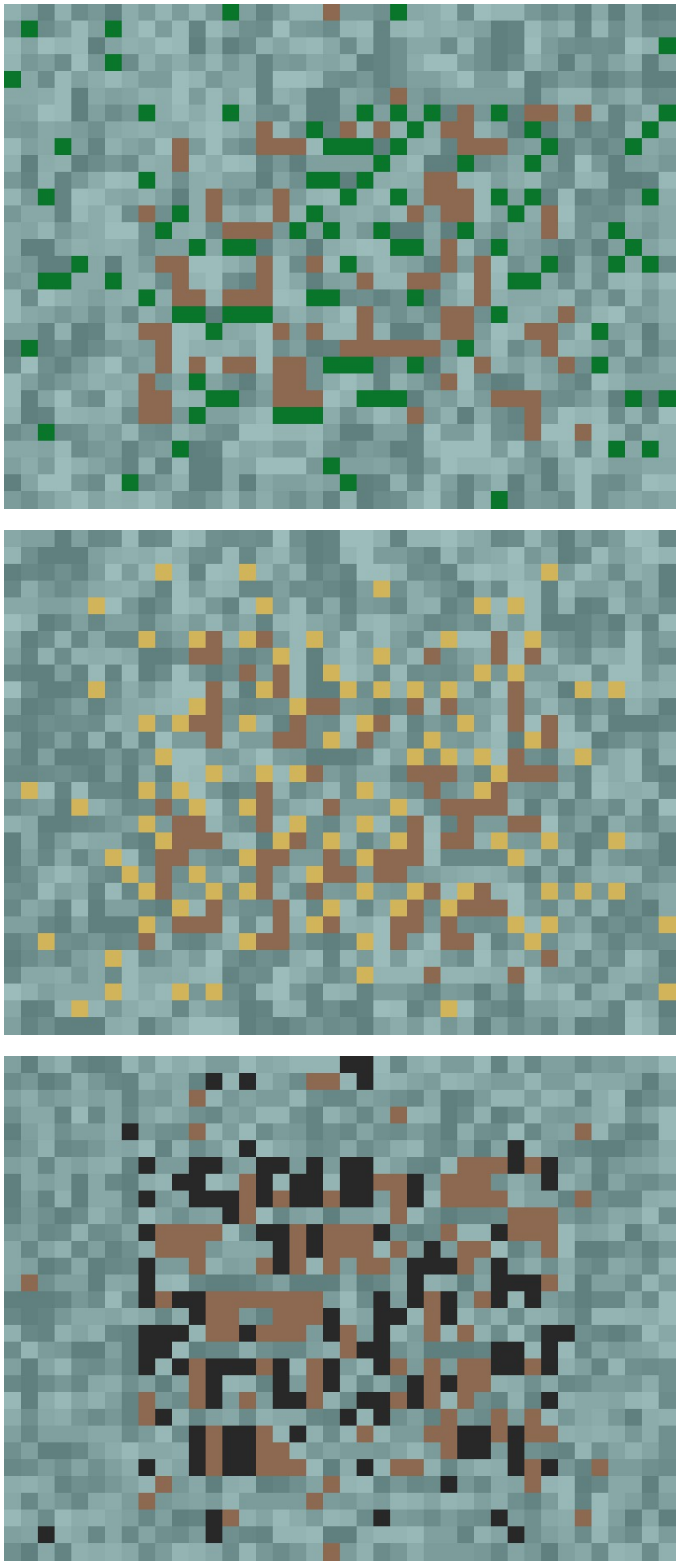

Figure 2: *Hex-chaos compositions. Top:* $x_1 = 2$ *(green),* $x_2 = 5$ *(brown). Middle:* $x_1 = 5$ *(brown),* $x_2 = 9$ *(yellow). Bottom:* $x_1 = 5$ *(brown),* $x_2 = f$ *(black).*

Conjecture A. $\Upsilon_{m,n}(x_1, x_2) = \Gamma_{m,n}(x_1, x_2)$.

Equivalently, if we denote the set of attained maximums by

$$\mathcal{M}_{m,n}(x_1, x_2) = \{M \in \mathcal{P}_{m,n} : min(\eta_{x_1}(M), \eta_{x_2}(M)) = \Gamma_{m,n}(x_1, x_2)\}$$

we have

Conjecture B. $\mathcal{M}_{m,n}(x_1, x_2) \cap \mathcal{E}_{m,n}(x_1, x_2) \neq \emptyset$.

Packing Problem Equivalences

For fixed m and n, using the convention $0 \leq x_1 < x_2 \leq f$, ostensibly there are 120 packing problems to consider. Our goal is to prove that there are really only thirteen such problems and thus, up to the equivalence relation defined below, only thirteen different kinds of hex-chaos compositions.

If $B = (b_{i,j})$ is $k \times \ell$ matrix of zeros and ones we define its *complement* $\chi(B)$ to be $J - B$ where J is the $k \times \ell$ matrix all of whose entries are ones. Complements establish four bijections of $\mathcal{P}_{m,n}$ with itself, χ_I, χ_R, χ_C, $\chi_B = \chi_R \circ \chi_C = \chi_C \circ \chi_R$ corresponding to the identity, complementing the row edges, complementing the column edges, and complementing both the row and column edges. That is,

$$\begin{aligned}
\chi_I(H \diamond V) &= H \diamond V \\
\chi_R(H \diamond V) &= \chi(H) \diamond V \\
\chi_C(H \diamond V) &= H \diamond \chi(V) \\
\chi_B(H \diamond V) &= \chi(H) \diamond \chi(V)
\end{aligned}$$

These bijections induced hex label permutations $\psi_I, \psi_R, \psi_C, \psi_B$ that are wholly determined by what happens to the edges of a "generic" cell. At this juncture it is convenient to change notation and use edge labels that reflect the ordering used for binary to hex conversion. We have

$$\begin{bmatrix} & e_1 & \\ e_2 & x & e_4 \\ & e_3 & \end{bmatrix} \longrightarrow \begin{bmatrix} & 1-e_1 & \\ e_2 & \psi_R(x) & e_4 \\ & 1-e_3 & \end{bmatrix}, \begin{bmatrix} & e_1 & \\ 1-e_2 & \psi_C(x) & 1-e_4 \\ & e_3 & \end{bmatrix}, \begin{bmatrix} & 1-e_1 & \\ 1-e_2 & \psi_B(x) & 1-e_4 \\ & 1-e_3 & \end{bmatrix}$$

whence running through the sixteen possibilities for the base two representation of x as $(e_1e_2e_3e_4)_2$ establishes ψ_R with $(e_1e_2e_3e_4)_2 \longrightarrow ((1-e_1)e_2(1-e_3)e_4)_2$ as

$$\psi_R = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & a & b & c & d & e & f \\ a & b & 8 & 9 & e & f & c & d & 2 & 3 & 0 & 1 & 6 & 7 & 4 & 5 \end{pmatrix},$$

and, similarly, we obtain

$$\psi_C = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & a & b & c & d & e & f \\ 5 & 4 & 7 & 6 & 1 & 0 & 3 & 2 & d & c & f & e & 9 & 8 & b & a \end{pmatrix},$$

$$\psi_B = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & a & b & c & d & e & f \\ f & e & d & c & b & a & 9 & 8 & 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 \end{pmatrix}.$$

If $M = (m_{i,j})$ is a $k \times \ell$ matrix we define the reflection of M about the *vertical* axis $\nu(M)$ to be the $k \times \ell$ matrix $(n_{i,j})$ where $n_{i,j} = m_{i,(\ell+1)-j}$, and the reflection of M about the *horizontal* axis $\mu(M)$ to be the $k \times \ell$ matrix $(u_{i,j})$ where $u_{i,j} = m_{(k+1)-i,j}$. These operations together with the matrix transpose operation $(M^t = (t_{i,j})$ where $t_{i,j} = m_{j,i})$ allow us to mimic the action of the dihedral group of order eight on an $m \times n$

pullback. We realize this group as four clockwise rotations of 0(90), 1(90), 2(90), and 3(90) degrees plus four reflections h, v, d and a about the horizontal, vertical, diagonal, and antidiagonal axes. We define

$$\begin{aligned}
\zeta_0(H \diamond V) &= H \diamond V \\
\zeta_1(H \diamond V) &= \nu(V^t) \diamond \nu(H^t) \\
\zeta_2(H \diamond V) &= (\mu \circ \nu)(H) \diamond (\mu \circ \nu)(V) \\
\zeta_3(H \diamond V) &= \mu(V^t) \diamond \mu(H^t) \\
\zeta_h(H \diamond V) &= \mu(H) \diamond \mu(V) \\
\zeta_v(H \diamond V) &= \nu(H) \diamond \nu(V) \\
\zeta_d(H \diamond V) &= V^t \diamond H^t \\
\zeta_a(H \diamond V) &= (\mu \circ \nu)(V^t) \diamond (\mu \circ \nu)(H^t)
\end{aligned}$$

The four bijections that involve transposes map $m \times n$ pullbacks to $n \times m$ pullbacks. We let π_* denote the induced hex label permutation for ζ_*. We know $\pi_0 = \psi_I$ is the identity permutation. To determine the other induced permutations, we again use our generic cell mapping notation, but now we must recognize that the cell containing hex label $x_{i,j}$ is being moved before it is relabeled. If we write $\tau(M)$ for M^t so the notation is consistent, then we can track label movement by making appropriate *index* calculations. For example, $\zeta_1((i,j)) = (\nu \circ \tau)((i,j)) = \nu((j,i)) = (j,(m+1)-i)$ tells us that the label in cell (i,j) will be moved to cell $((n+1)-j,i)$ and relabeled according to the permutation π_1 shown below. Because we know we are recovering the dihedral group, we only calculate ζ_1 and ζ_v:

$$\begin{bmatrix} & e_1 & \\ e_2 & x & e_4 \\ & e_3 & \end{bmatrix} \longrightarrow \begin{bmatrix} & e_2 & \\ e_3 & \pi_1(x) & e_1 \\ & e_4 & \end{bmatrix}, \begin{bmatrix} & e_1 & \\ e_4 & \pi_v(x) & e_2 \\ & e_3 & \end{bmatrix}$$

gives

$$\pi_1 = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & a & b & c & d & e & f \\ 0 & 2 & 4 & 6 & 8 & a & c & e & 1 & 3 & 5 & 7 & 9 & b & d & f \end{pmatrix},$$

$$\pi_v = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & a & b & c & d & e & f \\ 0 & 4 & 2 & 6 & 1 & 5 & 3 & 7 & 8 & c & a & e & 9 & d & b & f \end{pmatrix}.$$

Writing the non-identity permutations as products of disjoint cycles provides

$$\begin{aligned}
\psi_R &= (0\ a)(1\ b)(2\ 8)(3\ 9)(4\ e)(5\ f)(6\ c)(7\ d) \\
\psi_C &= (0\ 5)(1\ 4)(2\ 7)(3\ 6)(8\ d)(9\ c)(a\ f)(b\ e) \\
\psi_B &= (0\ f)(1\ e)(2\ d)(3\ c)(4\ b)(5\ a)(6\ 9)(7\ 8) \\
\pi_1 &= (0)(1\ 2\ 4\ 8)(3\ 6\ c\ 9)(5\ a)(7\ e\ d\ b)(f) \\
\pi_2 &= (0)(1\ 4)(2\ 8)(3\ c)(5)((6\ 9)(7\ d)(a)(b\ e)(f) \\
\pi_3 &= (0)(1\ 8\ 4\ 2)(3\ 9\ c\ 6)(5\ a)(7\ b\ d\ e)(f) \\
\pi_h &= (0)(1)(2\ 8)(3\ 9)(4)(5)(6\ c)(7\ d)(a)(b)(e)(f) \\
\pi_v &= (0)(1\ 4)(2)(3\ 6)(5)(7)(8)(9\ c)(a)(b\ e)(d)(f) \\
\pi_d &= (0)(1\ 2)(3)(4\ 8)(5\ a)(6\ 9)(7\ b)(c)(d\ e)(f) \\
\pi_a &= (0)(1\ 8)(2\ 4)(3\ c)(5\ a)(6)(7\ e)(9)(b\ d)(f)
\end{aligned}$$

The permutations ψ_R and ψ_C generate a subgroup of S_{16} isomorphic to the Klein four group using the familiar relations

$$\psi_R^2,\ \psi_C^2,\ \psi_R\psi_C = \psi_C\psi_R,$$

while the permutations π_1 and π_v generate a subgroup of S_{16} isomorphic to the dihedral group of order eight using the familiar relations

$$\pi_1^4,\ \pi_v^2,\ \pi_v\pi_1 = \pi_1^3\pi_v.$$

(Note that $\pi_1\pi_v = \pi_d$, $\pi_1^2\pi_v = \pi_h$ and $\pi_1^3\pi_v = \pi_a$.) However, the subgroup of S_{16} of order thirty two that these four permutations generate is *not* an internal direct product of the subgroup $<\pi_1, \pi_v>$ of order eight with the subgroup $<\psi_R, \psi_C>$ of order four because even though

$$\psi_R\pi_v = \pi_v\psi_R,\ \psi_C\pi_v = \pi_1\psi_C,$$

we also have

$$\psi_R\pi_1 = \pi_1\psi_C,\ \psi_C\pi_1 = \pi_1\psi_R.$$

To organize the calculation of equivalence classes of packing problems, we use a 4×8 table showing how each of the thirty two permutations in

$$<\pi_1, \pi_v, \psi_R, \psi_C> = \{\pi_1^i\pi_v^j\psi_R^k\psi_C^\ell : 0 \le i \le 3, 0 \le j, k, \ell \le 1\}$$

acts on an instance (x_1, x_2) — by letting the permutation act on each entry — in such a way that the first four columns show $m \times n$ equivalences and the last four show $n \times m$ equivalences. An example illustrates how this works.

$(0,1)$	π_0	π_2	π_h	π_v	π_1	π_3	π_d	π_a
ψ_I	$(0,1)$	$(0,4)$	$(0,1)$	$(0,4)$	$(0,2)$	$(0,8)$	$(0,2)$	$(0,8)$
ψ_R	(a,b)	(a,e)	(a,b)	(a,e)	$(5,7)$	$(5,d)$	$(5,7)$	$(5,d)$
ψ_C	$(5,4)$	$(5,1)$	$(5,4)$	$(5,1)$	$(a,8)$	$(a,2)$	$(a,8)$	$(a,2)$
ψ_B	(f,e)	(f,b)	(f,e)	(f,b)	(f,d)	$(f,7)$	(f,d)	$(f,7)$

Even though for the purpose of equivalence the pairs should show the smaller value first, the ordering of the pairs in the table is useful. For example the $(f,7)$ in the ψ_B row and π_3 column tells us that the mapping from $\mathcal{P}_{m,n}$ to $\mathcal{P}_{n,m}$ which takes $H \diamond K$ to $(\zeta_3 \circ \chi_B)(H \diamond K)$ will relocate all the 0's and 1's while simultaneously relabeling them as f's and 7's, respectively, because $(\pi_3 \circ \psi_B)(0) = f$ and $(\pi_3 \circ \psi_B)(1) = 7$. This, in turn, tells us $\Gamma_{m,n}(0,1) = \Gamma_{n,m}(7,f)$ or equivalently, if we can solve the $(0,1)$ packing problem for all problem instances then we can solve the $(7,f)$ packing problem for all problem instances. Thus, if we denote the equivalence class of packing problems for (x_1,x_2) by $[(x_1,x_2)]$, then the table reveals

$$\begin{aligned}[(0,1)] = \{&(0,1),(0,2),(0,4),(0,8),(1,5),(2,a),(4,5),(5,7),\\ &(5,d),(7,f),(8,a),(a,b),(a,e),(b,f),(d,f),(e,f)\}.\end{aligned}$$

Space prohibits showing the calculations for all thirteen equivalence classes. Instead, we show that the equivalence classes are not all the same size by providing

$(1,4)$	π_0	π_2	π_h	π_v	π_1	π_3	π_d	π_a
ψ_I	$(1,4)$	$(4,1)$	$(1,4)$	$(4,1)$	$(2,8)$	$(8,2)$	$(2,8)$	$(8,2)$
ψ_R	(b,e)	(e,b)	(b,e)	(e,b)	$(7,d)$	$(d,7)$	$(7,d)$	$(d,7)$
ψ_C	$(4,1)$	$(1,4)$	$(4,1)$	$(1,4)$	$(8,2)$	$(2,8)$	$(8,2)$	$(2,8)$
ψ_B	(e,b)	(b,e)	(e,b)	(b,e)	$(d,7)$	$(7,d)$	$(d,7)$	$(7,d)$

which yields

$$[(1,4)] = \{(1,4),(2,8),(7,d),(b,e)\},$$

and

$(3, c)$	π_0	π_2	π_h	π_v	π_1	π_3	π_d	π_a
ψ_I	$(3, c)$	$(c, 3)$	$(9, 6)$	$(6, 9)$	$(6, 9)$	$(9, 6)$	$(3, c)$	$(c, 3)$
ψ_R	$(9, 6)$	$(6, 9)$	$(3, c)$	$(c, 3)$	$(3, c)$	$(c, 3)$	$(6, 9)$	$(9, 6)$
ψ_C	$(6, 9)$	$(9, 6)$	$(c, 3)$	$(3, c)$	$(c, 3)$	$(3, c)$	$(9, 6)$	$(6, 9)$
ψ_B	$(c, 3)$	$(3, c)$	$(6, 9)$	$(9, 6)$	$(9, 6)$	$(6, 9)$	$(c, 3)$	$(3, c)$

which gives

$$[(3, c)] = \{(3, c), (6, 9)\}.$$

Table 1 lists the thirteen equivalence classes and their cardinalities. The choice of representative is lexicographical and classes are ordered lexicographically. With reference to Figure 2, we note that $(2, 5)$ belongs to $[(0, 7))]$, $(5, 9)$ belongs to $[(0, 3)]$ and $(5, f)$ belongs to $[(0, 5)]$. With reference to Theorem 3, $(5, f)$ belongs to $[(0, 5)]$.

Table 1: *Equivalence Classes of Hex-Chaos Compositions.*

equivalence class	cardinality
$[(0, 1)]$	16
$[(0, 3)]$	16
$[(0, 5)]$	4
$[(0, 7)]$	16
$[(0, f)]$	2
$[(1, 2)]$	16
$[(1, 3)]$	16
$[(1, 4)]$	4
$[(1, 6)]$	16
$[(1, b)]$	4
$[(1, e)]$	4
$[(3, 6)]$	4
$[(3, c)]$	2

Summary and Conclusions

We have described a generative art scheme incorporating chaotic one-dimensional cellular automata and hex labelings of the cells of a grid in order to evolve minimalist artworks called hex-chaos compositions. We then showed how this led to a suite of 120 packing problems which, thanks to some elementary group theory, could be organized into thirteen equivalence classes, thereby revealing there were exactly thirteen different types of hex-chaos compositions and presenting us with thirteen challenging packing problems to solve. To date we have solved seven of these problems.

References

[1] C. Cruz, M. Kirley, and J. Karakiewicz. “Generation and exploration of architectural form using a composite cellular automata.” *Third Australasian Conference, ACALCI 2017, Geelong VIC, Australia, January 31 - February 2, 2017, Proceedings*, M. Wagner *et al.* (eds.), LNAI 10142, Springer International Publishing, Cham, Switzerland, 2017, pp. 99–110.

[2] G. Greenfield. "Interleaved cellular automata, evolved art and packing problems." *2018 IEEE CEC Conference Proceedings*, to appear.

[3] S. Wolfram. “Universality and complexity in cellular automata.” *Phys. D Nonlinear Phenomena*, vol. 10, no. 1, 1984, pp. 1–35.