

# Combinatorial Figure Generation: An Algorithmic Approach

Conan Chadbourne

San Antonio, TX, USA; conan@conanchadbourne.com

## Abstract

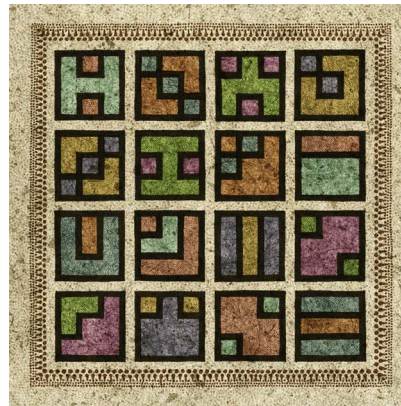
A system for producing small graphical symbols by partitioning symmetrical arrangements of tiles into figures composed of smaller shapes is described, and a procedure for constructing complete sets of all such figures for a given arrangement of tiles is presented. Several families of such tile arrangements are analyzed, and enumerations of the resulting figures are provided for arrangements of up to 16 tiles. The uses of graphical renderings of these figures in visual artworks is discussed, as are some methods for selecting subsets of the figures according to their intrinsic properties.

## Introduction

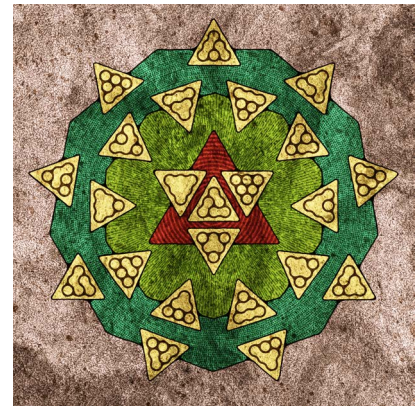
One of the interesting aesthetic questions that motivates me to produce mathematical art is the problem of how to represent abstract mathematical concepts in a purely visual way. To this end, I have developed a sort of visual language for representing mathematical objects, and one important part of the vocabulary of this visual language consists of various sets of small glyphs or other symbol-like forms. For several years I have used sets of such forms in my work, and these symbols have served a variety of different compositional and conceptual functions.



(a) *Intrinsic Regularity*  
(2015)



(b) *Abstruse Puzzle*  
(2019)



(c) *Obscure Instrument for Geomancy*  
(2019)

**Figure 1:** *Previous works using combinatorial symbols*

Sometimes, they function as semantically neutral tokens to represent a mathematical object or relationship, for example to illustrate some combinatorial structure or the elements of an abstract set. In this case, the intrinsic attributes of the symbols themselves are not significant to the conceptual basis of the work—they are simply placeholders and it is only important that they be distinct. For example, *Intrinsic Regularity* (Figure 1a), uses a set of seven circular symbols to represent the elements in the Steiner triple system on seven elements. This system is the unique way in which one can take subsets of three elements from a set of seven, such that any pair of elements occurs in exactly one subset. Since the elements can be drawn from *any* set of seven, their specific identities are unimportant; they only need to be distinguishable.

In other artworks, I use the attributes of the symbol itself to represent some property or characteristic of the mathematical system or structure that I am exploring in the image. In images that relate to a specific permutation group for example, I frequently use symbols as visual representations of the elements of the group, and I choose symbols that in some way indicate the order of the element or cycle structure of the permutation. Other examples are cases where combinatorial structures are represented by the qualities of the symbols. *Abstruse Puzzle* (Figure 1b) uses a grid of square symbols to represent a system of mutually orthogonal Latin squares of order 4. Each symbol has three attributes which, considered alone, form a Latin square—color pairs (red-cyan, blue-orange, green-magenta, and yellow-violet), symmetry (one or two reflection axes, aligned either parallel to the sides of the square or on the diagonal), and proportion of the symbol which is filled with a particular pattern ( $2/9$ ,  $1/3$ ,  $2/3$ , or  $7/9$ ). Each value of each attribute occurs exactly once in every row and column of the grid, thus creating a Latin square. The Latin square formed by considering any one of these attributes is orthogonal to that formed by either of the other two attributes.

Sometimes, the set of symbols itself is the conceptual core of the work—in particular the completeness of a set of symbols or their classification according to different attributes. *Obscure Instrument for Geomancy* (Figure 1c) takes this approach. All 23 possible symbols formed by partitions of a triangular array of six points are presented, arranged in concentric rings according to the subgroup of the full triangular group that leaves the configuration invariant.

Aesthetically, I see these sets of symbols as allied to various historical artistic and graphic traditions. In particular, a number of divination traditions have employed similar symbol sets, often with some underlying combinatorial structure. Perhaps the best known is the *I Ching*, or the Book of Changes, from China. Developed in the third century B.C.E.[4], this divination system uses symbols such as the ones below:



Graphically, the symbols consist of hexagrams of solid or divided lines, with the full set of symbols ranging over all of the 64 possible combinations. Other divination traditions have similar kinds of figures. Ascher [1] describes the combinatorial foundations of a number of other divination traditions.

### Aesthetic Criteria for Symbol Sets

There are a few criteria that I find important in creating a useful system of these symbols. While not every symbol system that I use necessarily meets all of these requirements, they are helpful guides for finding a set of symbols for a particular image or a particular mathematical concept that I want to explore.

#### *Properties as a system*

First, I generally like for the symbols within a system to follow some simple, usually combinatorial, set of rules, with some characteristics that are shared among all of the symbols and usually one other characteristic that varies between symbols. They need to be visually distinct, but similar enough that they are obviously related. Usually, there needs to be a finite, preferably reasonably small, set of possible symbols within the system. This is important in cases where the use of some minimal complete set is conceptually important to the artwork.

#### *Semantic ambiguity*

One of the qualities that I look for in these symbols is that they be semantically ambiguous. This is a bit of a fuzzy requirement. Of course, any set of symbols can be arbitrarily endowed with some semantic value—some sort of meaning or interpretation—through extrinsic associations, but I would like the intrinsic characteristics of the symbols to not obviously be indicative of any specific meaning. For example, I don't want them to seem like obvious stand-ins for numbers or words. At the same time, I like for them to look like they *could* have some interpretation, given some external, perhaps cultural, context.

The symbols in the *I Ching* are a good model for this. Each of the symbols has a specific interpretation within the tradition of Chinese divination, but a viewer unacquainted with this tradition would not necessarily have any basis on which to derive this interpretation solely from the graphic qualities of symbols themselves. And yet, as graphic figures they are evocative of a kind of ordered system, even for viewers who may not know the context.

This semantic ambiguity is particularly important in artworks like *Intrinsic Regularity*, mentioned earlier, where I want to use the symbols as placeholders for elements of an abstract set. This work presents a specific combinatorial structure, namely the Steiner triple system on a set of seven elements. In a typical mathematical treatment of the subject, such as in a textbook, the elements of the set might be represented by numerals or letters, with the understanding that this represents an arbitrary labeling of the set elements. This can sometimes be a bit misleading, however, since numerals and letters bring along semantic qualities and implications, such as a natural ordering, that do not exist in the set itself. I want to be able to create visual representations of structures like this which avoid this dilemma.

### ***Graphical Qualities***

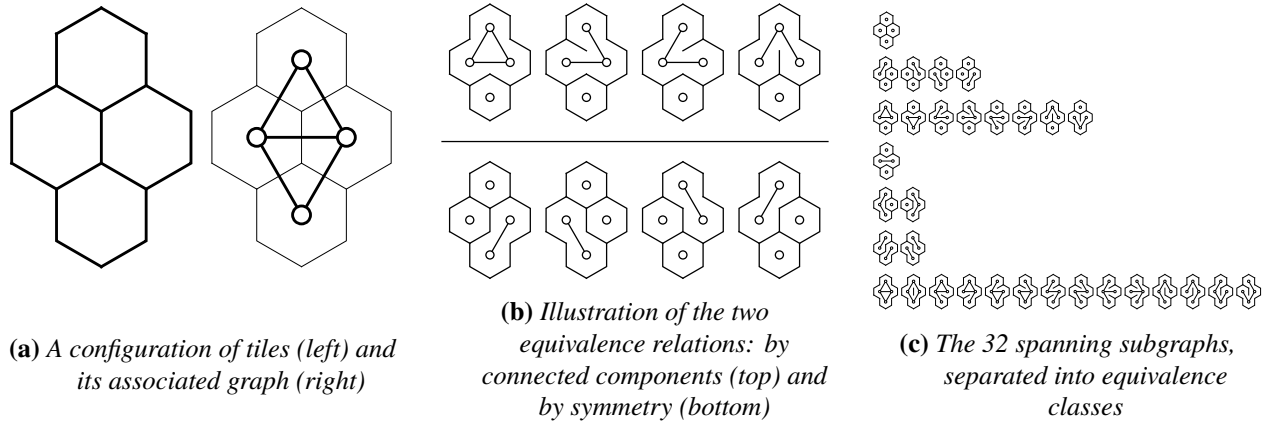
Finally, I like to choose symbols that have some particular qualities as graphic elements in a composition. These are generally subjective design choices based on the requirements of any given composition. I often seek sets of symbols that have a uniform external shape, allowing them to function as interchangeable graphic objects. I prefer symbols that can be drawn as a single connected form, and which can be rendered in a single color. I generally like the symbols not to imply or require any obvious orientation, so that they can be rotated or reflected arbitrarily to fit the needs of the composition that I am using them in. Finally—and most subjectively—I look for sets of symbols that I find visually interesting, that have some sense of balance or proportion that is appealing.

## **Generating Symbols Using Shape Partitions**

Because these sets of symbols are so useful to my artistic practice, I began exploring various approaches to creating new sets of them for use in my work. One approach that I found particularly fruitful was to take a fixed, usually symmetrical, configuration of tiles and construct all of the possible partitions of the configuration into subsets of adjacent tiles. The square symbols in *Abstruse Puzzle*, for example, are created by taking partitions of the set of nine square tiles arranged in a 3x3 grid; the triangular symbols in *Obscure Instrument for Geomancy* are created by taking partitions of the set of six hexagonal tiles, arranged in a triangular grid. I have also applied this approach to build symbol sets using other shapes for the tiles, such as isosceles right triangles, equilateral triangles, and rhombuses.

In his 2019 Bridges paper, James Mai [3] describes a similar system for partitioning a grid of points into separate subsets of adjacent points, for three specific cases with 6 or 9 points. He describes these as “shape partitions” when the points are arranged in a grid configuration, and “ring partitions” when the points are arranged at the vertices of a regular polygon. I sought to generalize this approach to additional configurations, and to look for a systematic method for the construction of all such partitions for any given configuration. I adopt his “shape partition” nomenclature throughout, to refer to all configurations under consideration, and restate the rules that he set down for this system, translating them a bit to apply to these more general configurations (and also to refer to tiles rather than points).

For a given configuration of tiles, we will consider a *shape partition* of the configuration to be a way of connecting tiles into component shapes such that: (1) Every tile belongs to one and only one component shape; (2) Each component shape is composed only of contiguous tiles—i.e. only adjacent tiles can be connected; (3) If any two adjacent tiles belong to the same component shape, then they are connected; (4) If any two such partitions are congruent when rotated or reflected, they are not considered distinct.



**Figure 2:** Procedure for finding shape partitions, applied to a patch of four hexagonal tiles ( $R_2$ )

### Formal Treatment of Shape Partitions

We can use the techniques of graph theory to formalize the notion of shape partitions, and to analyze them in a systematic way. For simplicity, we will consider a *tile* to be any planar region that is homeomorphic to a closed, 2-dimensional disk; it will therefore be compact and simply connected. We will define a *tile configuration* to be a finite collection of such tiles such that the intersection of any two tiles is either the empty set, a finite set of discrete points, or a single edge. Two tiles are *adjacent* if their intersection is a single edge. We will also require the union of all of the tiles in the set to be connected. We begin by constructing a graph model for the configuration of tiles.

Starting with a configuration of  $n$  tiles  $T_1, T_2, \dots, T_n$  along with the group  $\Gamma$  of geometric symmetries (i.e. isometries of the plane) that leave the tile configuration invariant, we construct a graph  $G = (V, E)$  by associating a vertex  $v_i \in V$  to each tile  $T_i$  and connecting two vertices by an edge if the associated tiles are adjacent. This association is illustrated in Figure 2a. We also construct the associated group  $\Gamma_E$ , which is isomorphic to  $\Gamma$ , and which represents the action of  $\Gamma$  as a group of permutations on the set of edges in  $G$ . In this model, a shape partition will be a subgraph of  $G$ , the component shapes will be the connected components of the subgraph, and the "connections" between tiles within the component shapes will correspond to the edges of the subgraph. In particular, a shape partition will be a *spanning subgraph* of  $G$ , meaning it will contain all of the vertices in  $G$  and a subset of the edges in  $G$ . This follows from our first two rules, that all tiles must belong to one and only one component shape, and that only adjacent tiles may be connected.

We let  $\mathcal{H}(G)$  denote the set of all spanning subgraphs of  $G$ , i.e.  $\mathcal{H}(G) = \{(V', E') \mid V' = V, E' \subseteq E\}$ . For every edge  $e_i \in E$ , either  $e_i \in E'$  or  $e_i \notin E'$ , so  $|\mathcal{H}(G)| = 2^m$ , where  $m = |E|$  is the number of edges in  $G$ . We proceed by constructing two equivalence relations on  $\mathcal{H}(G)$ . First, we note that if a spanning subgraph  $H \in \mathcal{H}(G)$  has  $k$  connected components it will partition the vertex set  $V$  into  $k$  disjoint, non-empty subsets  $V_1, V_2, \dots, V_k$ , where:

$$\bigcup_{i=1}^k V_i = V \quad \text{and} \quad V_i \cap V_j = \emptyset \quad \text{for} \quad i \neq j$$

We consider any two subgraphs equivalent if they produce the same partition of the vertex set. This is illustrated in the upper half of Figure 2b, which shows an equivalence class of four subgraphs whose connected components all have the same vertices.

The third rule for shape partitions—that if any two adjacent tiles belong to the same component shape, then they are connected—gives us a well-defined way to choose a representative of the equivalence class. In particular, we want to avoid "internal boundaries," such as the ones in the three rightmost graphs in the upper

part of Figure 2b. This is achieved by selecting the member of the equivalence class with the greatest number of edges, or, equivalently, by taking the sum of the induced subgraphs for each of the connected components:  $\langle V_1 \rangle + \langle V_2 \rangle + \cdots + \langle V_k \rangle$ .

Our fourth rule for shape partitions—that they are not considered distinct if they are congruent when rotated or reflected—leads to the second equivalence relation on  $\mathcal{H}(G)$ . Specifically, for any two subgraphs  $H_1=(V, E_1)$ ,  $H_2=(V, E_2) \in \mathcal{H}(G)$ ,  $H_1 \equiv H_2$  if there is some  $\gamma \in \Gamma_E$  such that  $\gamma(E_1) = E_2$ . This is illustrated in the lower half of Figure 2b. The four graphs shown can all be transformed into one another by either reflection or rotation. Harary [2] showed that the number of distinct equivalence classes relative to a group  $\Gamma_E$  can be determined using Pólya-Redfield enumeration techniques as follows: We form the *cycle index polynomial*

$$\mathcal{Z}(\Gamma_E; a) = \frac{1}{|\Gamma_E|} \sum_{\gamma \in \Gamma_E} \prod_{k=1}^d a_k^{j_k(\gamma)}$$

where  $j_k(\gamma)$  is the number of cycles in  $\gamma$  of length  $k$ . The number of equivalence classes under the symmetries of  $\Gamma_E$  is then given by:  $\mathcal{Z}(\Gamma_E; 2)$ , that is, by setting  $a_1 = a_2 \dots a_d = 2$ .

## Enumeration, Construction and Classification of Shape Partitions

There are three natural questions that we may want to ask about any given system of symbols. The first question is about enumeration: Exactly how many symbols are there in the system? The second question is about construction: How can we produce all of these symbols? The third question is about classification: What are some attributes or properties of the symbols that we can use to divide up the set of symbols into smaller sets that share some attribute? We explore these questions in the case of the system of shape partition symbols by looking at some specific examples.

### Graph Families to be Studied

The procedure outlined above for finding shape partitions is quite general—we can apply it to any configuration of adjacent tiles and the group of symmetries that the configuration is invariant under. We consider here several parametrically specified families of planar graphs that have nice symmetrical embeddings which can be associated with symmetrical configurations of tiles. Table 1 lists these families and some of their properties.

One natural way to choose a configuration is to take a connected patch of tiles from one of the three regular tessellations by squares, triangles or hexagons. This has the advantage that all of the tiles are identical. In this case, the possible rotational symmetries of the configuration are limited to 2-fold and 4-fold rotations in the case of square tiles, and 2-fold, 3-fold and 6-fold rotations in the case of triangles or hexagons. This means that the symmetry groups leaving the configuration invariant will be isomorphic to either the cyclic groups  $C_2$ ,  $C_3$ ,  $C_4$ , and  $C_6$ , or the dihedral groups  $D_1$ ,  $D_2$ ,  $D_3$ ,  $D_4$ , and  $D_6$ . The graph families from Table 1 that correspond to configurations of square tiles are  $P_n$ ,  $L_n$  and  $G_p$ . Those corresponding to configurations of hexagonal tiles are  $P_n$ ,  $T_p$  and  $R_p$ . Finally,  $H_p$  corresponds to configurations of triangular tiles.

Another natural way to choose configurations is to arrange tiles symmetrically around a fixed point. In this case, the tiles need not be regular polygons, and may not necessarily all be identical. The graph families in Table 1 that correspond to such configurations are the cycle graphs  $C_n$  and wheel graphs  $W_n$ .

### Enumeration

It is worth noting that while the problem of the enumeration of the complete sets of shape partitions is the more interesting—and difficult—question, mathematically speaking, the problem of actually constructing complete sets of figures is the more meaningful one for the purposes of using them in artworks. Additionally, if we can algorithmically generate all of the shape partitions for a given system, then we can solve the enumeration problem for that system by simply counting how many solutions our algorithm finds. This

**Table 1:** *Parametrically Defined Families of Graphs to be Studied*

Graph Family	Examples	Vertices $n =  V $	Edges $m =  E $	Symmetry $\Gamma$
Path Graphs $P_n$		$n$	$n - 1$	$D_1$
Cycle Graphs $C_n$		$n$	$n$	$D_n$
Wheel Graphs $W_n$		$n$	$2(n - 1)$	$D_{n-1}$
Ladder Graphs $L_n$		$n$	$3n - 2$	$D_2$
Triangular Grid Graphs $T_p$		$\frac{p(p+1)}{2}$	$\frac{3p(p-1)}{2}$	$D_3$
Square Grid Graphs $G_p$		$p^2$	$2p(p - 1)$	$D_4$
Rhombus Graphs $R_p$		$p^2$	$(3p - 1)(p - 1)$	$D_2$
Hexagonal Grid Graphs $H_p$		$p^2$	$\frac{3p(p-1)}{2}$	$D_3$

may not be the most satisfying approach from a mathematical perspective, since it only works for smaller systems—the number of calculations required to algorithmically produce these figures becomes impractical very quickly for larger systems—but, in practice, the small systems are the ones which are more applicable for producing graphic symbols.

It is not clear whether it is possible to find a general method for enumerating shape partitions in most cases, but there are two cases that can be solved easily. The first case is when the graph for the system is a *tree*, that is when it contains no cycles. In this case, every spanning subgraph uniquely partitions the vertex set, so the number of shape partitions is equal to  $\mathcal{Z}(\Gamma_E; 2)$ . In Table 1, the only family of graphs that this applies to are the path graphs  $P_n$ . The other case is when the graph is a cycle. In this case, removing any single edge turns the graph into a tree, so the number of shape partitions is  $\mathcal{Z}(\Gamma_E; 2) - 1$ . The family of cycle graphs  $C_n$  in Table 1 represent this case. For all other families of graphs, we enumerate the shape partitions by algorithmically generating all of them, as described earlier. Table 2 shows the number of shape partitions for each of the graph families in Table 1 with up to 16 vertices.

The number of shape partitions grows very quickly with the number of edges in the graph. This makes intuitive sense—if shape partitions are formed by connecting adjacent tiles in a configuration, more adjacencies in the configuration means more different ways of connecting tiles. For example, if we consider the three different grid configurations of 9 tiles—using triangles, squares, and hexagons, corresponding to graphs  $H_3$ ,  $G_3$ , and  $R_3$ , respectively—the number of shape partitions grows from 96, for  $H_3$  with 9 edges, to 216 for  $G_3$  with 12 edges, to 600 for  $R_3$  with 16 edges. (In his 2019 paper, James Mai lists 593 of the shape partitions for the  $R_3$  system; the seven additional graphs are:

**Construction**

The procedure described earlier suggests a template for a simplistic, if inefficient, algorithm for finding all of the shape partitions for a given configuration. First, the configuration is modeled by its adjacency graph, and the list of all possible spanning subgraphs is produced. Then, this list is divided into classes according to the two equivalence relations described—i.e. by the vertex partition induced by the subgraph, and by equivalence under symmetry. Finally, a representative is chosen from each equivalence class by selecting a

**Table 2:** Calculated number of shape partitions for the families of graphs in Table 1. (Empty entries correspond to cases where a configuration does not exist for the given number of vertices.)

$n$	$P_n$	$C_n$	$W_n$	$L_n$	$T_p$	$G_p$	$R_p$	$H_p$
3	3	3			3			
4	6	5	7	7		5	7	4
5	10	7	13					
6	20	12	23	29	23			
7	36	17	46					
8	72	29	88	143				
9	136	45	186			216	600	96
10	272	77	395	780	1028			
11	528	125	880					
12	1056	223	1989	4546				
13	2080	379	4644					
14	4160	686	10934	27269				
15	8256	1223	26210		221608			
16	16512	2249	63319	166070		212987	*	33432

\* Calculations for the  $R_4$  system have not been completed at this time.

graph with the maximal number of edges. For this paper, this approach is implemented using *Mathematica*, and the subgraphs are represented by integers ranging from 0 to  $2^m$ , with the binary digits of each integer representing the presence or absence of a particular (indexed) edge of the graph.

### Classification

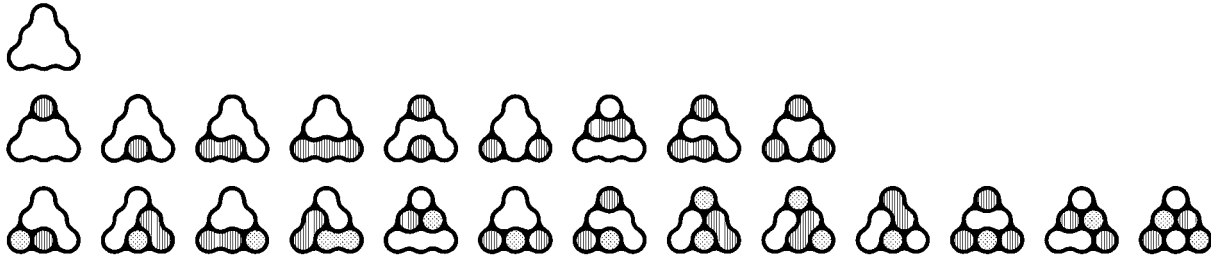
The complete sets of shape partitions for some of the simpler graphs are reasonably small, and these sets can be used directly in artworks. However, as the size and complexity of the graph increases, the number of shape partitions quickly grows to unmanageable levels. It therefore becomes useful to be able to select subsets of the shape partitions based on some intrinsic criteria. Mathematically, this translates to finding additional equivalence relations on the set of shape partitions, and examining the equivalence classes themselves as potentially interesting subsystems. We present here a few different attributes of these symbols that can be used for this classification process.

One of the most natural ways of classifying shape partitions is by associating them to their corresponding integer partition—i.e. the numbers of tiles in each of the component shapes, which will add up to the total number of tiles in the configuration. Another natural classification is according to the geometric symmetries of the partition—i.e. the subgroup of the symmetries of the configuration that leave the partition invariant. James Mai discusses both of these classification schemata in some depth in his paper. We present here just one example to illustrate the effectiveness of this approach. The  $4 \times 4$  square grid system has 212,987 shape partitions, making it entirely too large for the purposes of creating graphic symbols, but we can ask, for example, how many of these partitions are there that divide the grid into two parts of equal area, corresponding to the integer partition  $8+8$ . There are only 13 such partitions, shown in Figure 3. This is a much more manageable subsystem for producing symbols to use in artworks.



**Figure 3:** The 13 Partitions of the  $G_4$  tile grid into two pieces of equal area

Another intrinsic property that offers some possible utility in classifying shape partitions is the minimum number of colors required to color the components of the partition such that no two adjacent components are the same color. Mathematically, this is the chromatic number  $\chi$ , of the adjacency graph of the partition components. Figure 4 shows the 23 shape partitions of the  $T_3$  graph, sorted by chromatic number.



**Figure 4:** The 23 Symbols used in Figure 1a (shape partitions for graph  $T_3$ ), sorted by chromatic number  $\chi$

Since we know that the configuration for any shape partition is planar, the Four Color Theorem tells us that  $\chi$  is equal to 1,2,3, or 4, and  $\chi = 1$  only for the partition consisting of a single connected component. This means that effectively there are only three “bins” that this sorts most of the shape partitions into. For very large systems, this property alone doesn’t produce a fine enough classification to select reasonably small sets, but it can be useful in combination with some other classification, such as the ones described above. A related property, the number of *distinct* colorings for a given number of colors can also be used, and may produce finer classifications for some systems.

### Graphical Rendering of Shape Partitions

Once the shape partitions for a system are determined, these can be rendered as graphical symbols for use in compositions. There is considerable artistic freedom in the design of these renderings—this is where the mathematical problem of finding shape partitions turns into the design problem of using them to create visually interesting and evocative symbols. The shapes of the tiles can be varied, or the tiles themselves can be dispensed with altogether and the the vertices of the underlying graph can be used. The visual indication of connections between adjacent tiles or vertices can be varied as well. A single combinatorial system of shape partitions can be used to create many different sets of graphical symbols, to fit the needs of the composition in which they will be used.

### Conclusions and Future Work

One of the limitations of the algorithm used here for finding shape partitions is that it is extremely inefficient. The number of subgraphs that must be checked grows exponentially with the number of edges (or connections between tiles). This makes finding shape partitions for larger systems impractical. A possible area for future work would be to seek a more efficient algorithm, perhaps using a more constructive approach—such as somehow generating new shape partitions recursively from known ones—rather than the reductive one used here which requires testing all possible subgraphs. Another area for future work would be to identify additional intrinsic properties of the shape partitions that could be used for producing finer classifications of the shape partitions within a system. It might also be interesting to apply the techniques developed here to other families of planar (or even non-planar) graphs.

### References

- [1] M. Ascher. *Mathematics Elsewhere: An Exploration of Ideas Across Cultures*, Princeton University Press, 2002.
- [2] F. Harary. “On the Number of Dissimilar Line-Subgraphs of a Given Graph.” *Pacific Journal of Mathematics*, vol. 6, 1956, pp. 57–64.
- [3] J. Mai. “Shape-Partitions: New Elements, New Artworks.” *Bridges Conference Proceedings*, Linz, Austria, Jul. 16–20, 2019, pp. 171–178.
- [4] J. Needham. *The Shorter Science and Civilization in China*, Cambridge University Press, 1978.