Animated Motions of Exceptional Flexible Instances of Generically Rigid Graphs

Georg Grasegger[1], Jan Legerský[2] and Josef Schicho[2]

[1]Johann Radon Institute for Computational and Applied Mathematics (RICAM),
Austrian Academy of Sciences (ÖAW), Austria; georg.grasegger@ricam.oeaw.ac.at
[2]Research Institute for Symbolic Computation (RISC),
Johannes Kepler University Linz, Austria; jan.legersky@risc.jku.at, josef.schicho@risc.jku.at

Abstract

In this paper we use a recent algorithm for finding flexible realizations of graphs. In particular we consider generically rigid graphs that have special non-generic instances in which edge lengths can be found such that we get a continuous motion. The graphs we present have a symmetric structure and allow flexible unit distance realizations.

Introduction

Consider the following object (Figure 1) constructed from wooden sticks and screws. As described in [1] the object is rigid for generic edge lengths. This means fixing one edge (wooden stick) in the coordinate system, all other coordinates are determined by the fact that the sticks cannot change length. Mathematically we
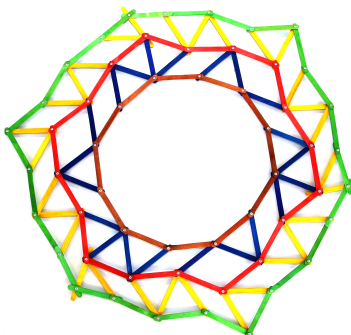
Figure 1: *A rigid graph constructed with wooden sticks and screws.*

describe the object as a graph $G = (V, E)$, where V is the set of vertices, in our case the screws, and E is the set of edges connecting two vertices. The length of the edges we describe by a function $\lambda : E \longrightarrow \mathbb{R}$. Maybe there exist coordinates for the vertices such that the graph can be drawn in the plane with the respective edge lengths. We call such coordinates a realization of the graph which is compatible with λ. We call a graph generically rigid if there are only finitely many realizations up to translations and rotations for edge lengths that are generic enough, i.e. avoiding cases where edge lengths fulfill specific algebraic equations. These sets of cases are, however, of measure zero. This means that for most choices of edge length there are no motions possible. Generically rigid graphs have been classified by [6, 5, 4].

A graph that is not rigid is called flexible. Obviously such a graph allows some motions in general. They are studied in the frame of linkages. As the definition suggests even for generically rigid graphs there are very particular edge lengths for which there are infinitely many realizations. They are of interest to us on
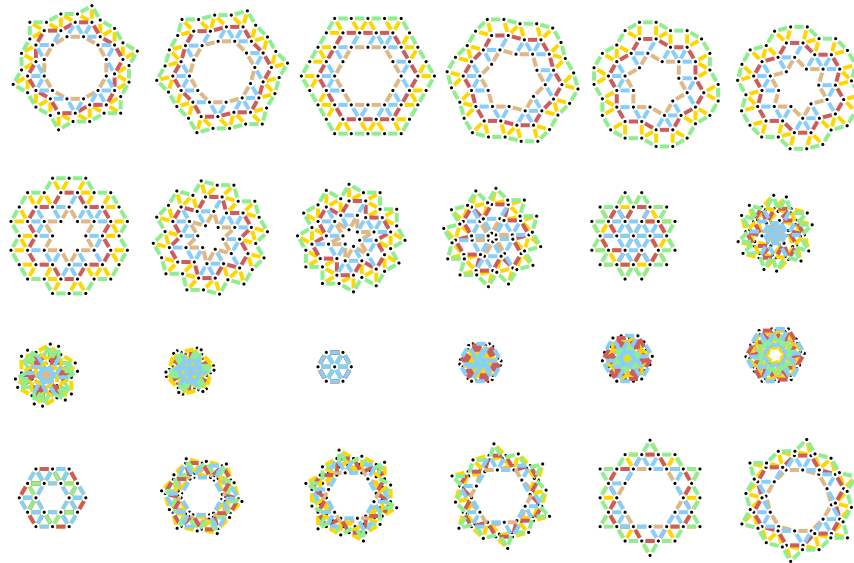
Figure 2: *Animated graph of Figure 1 with unit distance lengths*

the one hand because they are not so easy to find and on the other hand because they give nice rotationally symmetric animations for some graphs. In this paper we therefore analyze special edge lengths for a class of generically rigid graphs which allow a motion. A first example is the graph in Figure 1 which yields the following motion when all the edge length are chosen to be equal (Figure 2).

Most of the examples in this paper will have such unit distance realizations. The methods we use come from a recent paper [2] which finds flexible realizations for any graph by a coloring of their edges. These colorings and the construction of the respective realization and the motions are described in the next section. Afterwards we show how to use this method to get interesting animations. Within the paper we show the motions by consecutive figures. All the animations can be found in the supplementary material.

Colorings and Motions

In this section we describe the method from [2] that shows how to find a flexible instance for a generically rigid graph with help of a coloring of the edges in two colors.

Definition 1. *Let $G = (V, E)$ be a graph. A map $\delta : E \longrightarrow \{\text{blue}, \text{red}\}$ is called NAC-coloring if δ is surjective and every cycle in the graph is either monochromatic or has at least two edges in every color.*

Equivalently we can say there is "No Almost Cycle" (NAC), where an almost cycle has all but one edge in one color. Figure 3 shows a non-example and a correct example for a NAC-coloring. We can easily see that three-cycles in the graph always have to be monochromatic by definition.
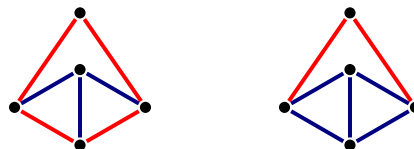

Figure 3: *The coloring on the left is not a NAC-coloring. The one on the right is a NAC-coloring.*

Given a graph with a NAC-coloring we take equivalence classes of the vertices with respect to two equivalence relations. Two vertices are in the same red equivalence class if there is a monochromatic red path between these two vertices. Similarly they are in the same blue equivalence class if there is a monochromatic blue path between them. See Figure 4 for an example.

$R_0 = \{1,2,3\}$	$B_0 = \{1,4\}$
$R_1 = \{4,5,6\}$	$B_1 = \{2,5\}$
	$B_2 = \{3,6\}$
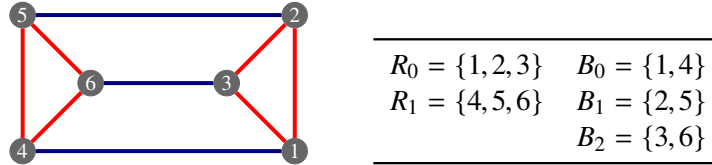
Figure 4: *A graph with NAC-coloring and the corresponding equivalence classes.*

Now we place the vertices in a coordinate system by indexing the equivalence classes. All vertices in the i-th red equivalence class have y-coordinate i and all vertices in the j-th blue equivalence class have x-coordinate j. This means we place the vertices in a grid. Let $R_0, \ldots, R_m$ be the red equivalence classes and $B_0, \ldots, B_n$ be the blue equivalence classes. We define the realization $\rho : V \longrightarrow \mathbb{R}^2$ by $\rho(v) = (i,j)$ if $v \in R_i \cap B_j$. By this method it might happen that some vertices are placed at the same coordinates. For some graphs this is unavoidable. Of course this depends on the NAC-coloring we chose. In Figure 5 a graph with three different NAC-colorings and the corresponding flexible realizations is shown. Two of the NAC-colorings yield overlapping vertices whereas the last one maps all vertices to different coordinates. The flexibility that arises is also indicated in the figure. In fact for every angle α we can define $\rho_\alpha : V \longrightarrow \mathbb{R}^2$ by $\rho_\alpha(v) = (i,0) + j \cdot (\cos\alpha, \sin\alpha)$. All ρ_α yield the same edge lengths for our given graph and by traversing α we get a continuous motion. In this paper all of our examples have a NAC-coloring that does not overlap
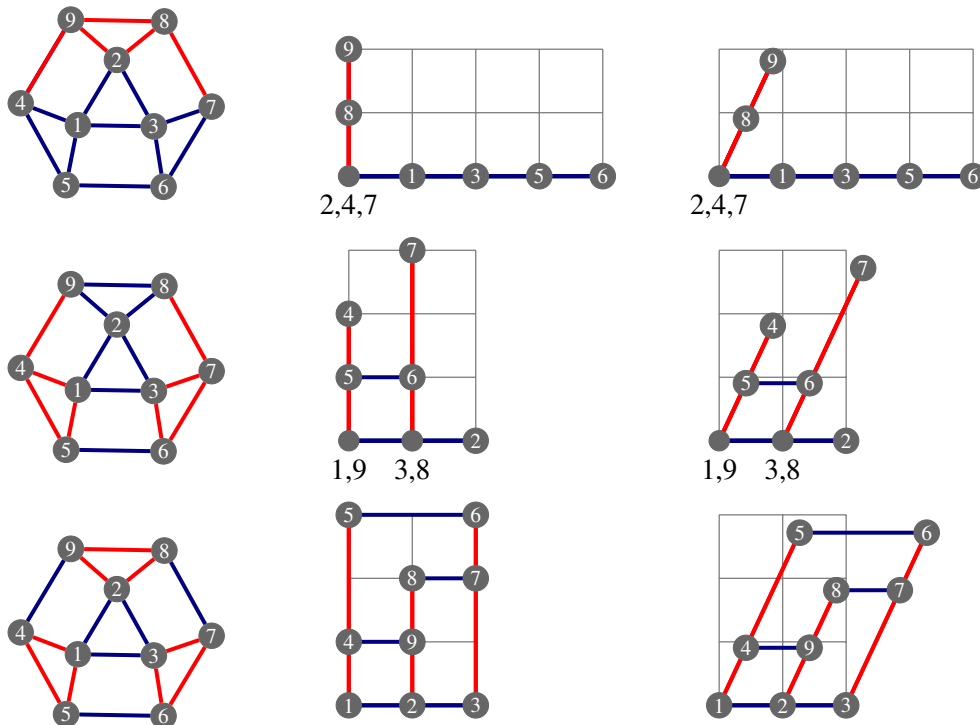

Figure 5: *Different NAC-colorings and the corresponding flexible realizations shown in two angles.*

vertices. Motions with non-overlapping vertices are studied in [3].

This basic grid construction also leads to degenerate triangles. Since a triangle has to be monochromatic always, the vertices will be placed on a line. This however can be avoided be using a more sophisticated grid structure. Let $a_1, \ldots, a_n$ and $b_1, \ldots, b_m$ be pairwise distinct coordinates different from $(0,0)$ and let $a_0 = b_0 = (0,0)$. Then we define realizations $\rho'_\alpha : V \rightarrow \mathbb{R}^2$

$$\rho'_\alpha(v) = a_j \cdot \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} + b_i \,,$$

where i and j are such that $v \in R_i \cap B_j$. These realizations yield again the same edge lengths for our graph, and hence we have a continuous motion by traversing α. We regard this construction to be kind of a zig-zag grid (see Figure 6).
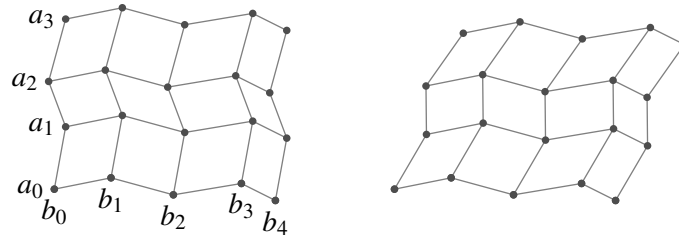

Figure 6: *Zig-Zag grid construction.*

Let us again consider the graph from Figure 5 and the third NAC-coloring. The first thing we would like to do is to avoid degenerate triangles. Hence, we move b_1 upwards to resolve the triangle $1, 2, 3$. Adjusting the a_i we also separate the red triangles. The animation `GridTransformation.svg` in the supplementary material shows how to find the zigzag grid. Indeed by choosing

$$a_0 = (0,0), \qquad a_1 = \left(-\sqrt{3}/2, 1/2\right), \qquad a_2 = \left(\sqrt{3}/2, 1/2\right), \qquad a_3 = (0,-1),$$
$$b_0 = (0,0), \qquad b_1 = \left(1, \sqrt{3}\right), \qquad b_2 = (2,0),$$

we get the following flexible realization (Figure 7). The flexibility comes from rotating the outer triangles around the vertices of the inner triangle respectively.
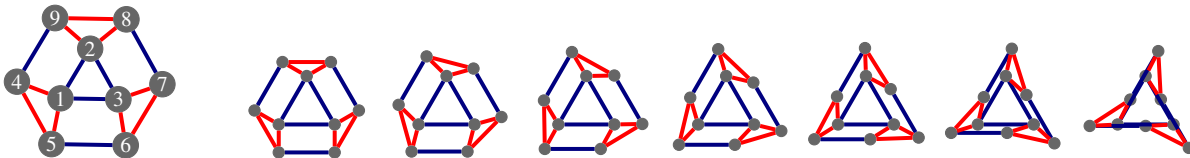
Figure 7: *A flexible realization with distinct coordinates for all vertices.*

This method can be applied to every graph, presuming we can find a NAC-coloring. However, in this paper we show some rotationally symmetric graphs whose motions yield interesting animations.

Special Symmetric Graphs and their Motions

Using the construction from the previous section we can find similar motions as in Figure 7 for larger graphs. All the graphs are generically rigid. By using NAC-colorings we find special instances for the edge lengths

such that we get an exceptional flexibility. Indeed in all the examples we have unit distance edges. The NAC-colorings have been found manually. There isn't any efficient algorithm so far for finding all NAC-colorings of a graph. In our case we are only interested in a single symmetric one, so it can be done by hand. This is simplified by the fact that all edges of a triangle have to be of the same color. Furthermore, since we do not want vertices to coincide, we can assume that in quadrilaterals opposite edges have the same color (compare cycle (1,2,9,4) in the first NAC-coloring of Figure 5).

Figure 8 shows the NAC-colorings of the motions presented in Figure 2 and Figures 9–14. These figures show images of the animation at equidistant time steps. The animations themselves are provided in the supplementary material of this paper as SVG (Scalable Vector Graphics) files. Hence, they can be opened by most standard internet browsers. In Figure 9 we show the animation using NAC-colorings whereas in the following figures the colors are used to get more visual insight in the motion. The SVG animations in the supplementary material are provided both with NAC-colorings and with the additional colorset.
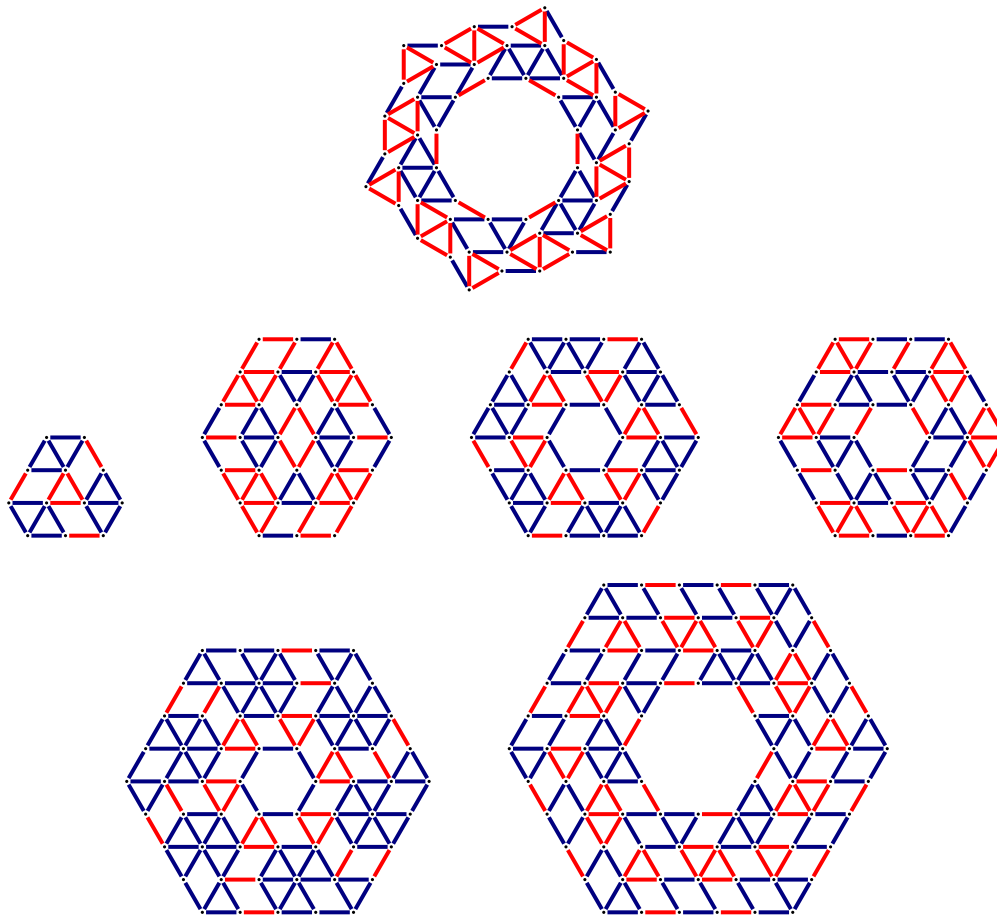

Figure 8: *NAC-colorings for graphs used in Figure 2 and Figures 9–14.*

Finding suitable a_i and b_j is the more difficult task. All the examples here have a realization where the vertices are indeed placed on a triangular grid with equilateral triangles (compare realizations in Figure 8). Having this in mind the order of the equivalence classes and the values for the a_i and b_j can be rather easily computed. The zig-zag grid construction finally provides a parametrization of the coordinates of the vertices and hence, for the motion.

Note that the motion is usually not centered by the symmetry of the figure but it keeps some triangle

fixed. This is because the baseline of the grid (i.e. the coordinates b_j) never moves. An exception is Figure 11 where the central hexagon is fixed. This is due to the chosen NAC-coloring, i.e. the hexagon is monochromatically colored and the equivalence class is chosen to be the first one. Its vertices are therefore placed on the non-moving baseline of the grid.
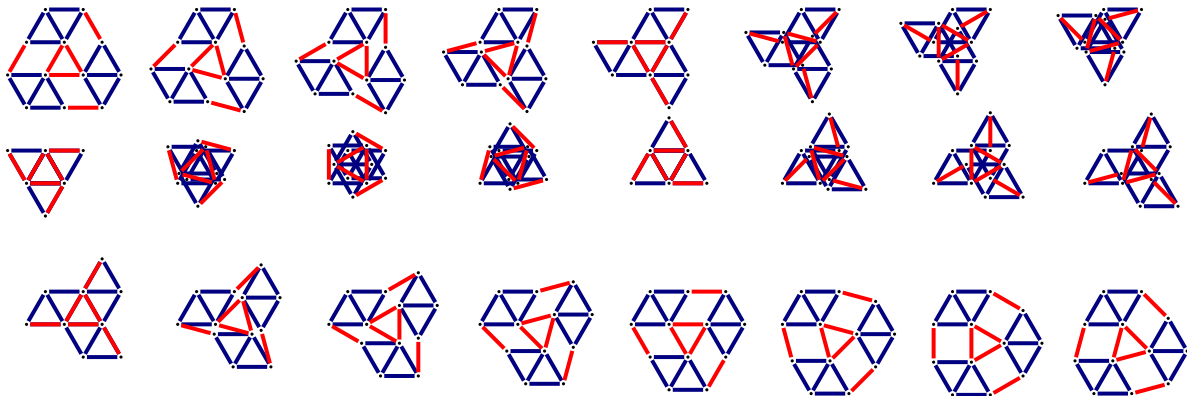
**Figure 9:** *Colors represent the NAC-coloring.*

Summary and Conclusions

We have shown how to find flexible instances of special symmetric generically rigid graphs. Furthermore, we explained how to theoretically construct the motion and create animations thereof. Figure 1 shows a rigid construction. By adjusting edge lengths according to the procedure presented in this paper we can make it flexible. In reality, however, a simple construction would yield several self collisions. Hence, the physical object only allows a part of the motion to be shown. In general it is not an easy task to provide collision-free layouts. This is subject to further research.

Acknowledgements

References

[1] G. Grasegger. "Realizations and Constructions of Minimally Rigid Graphs." In *Bridges Stockholm 2018 Conference Proceedings*, pages 239–242, Phoenix, Arizona, 2018. Tessellations Publishing.

[2] G. Grasegger, J. Legerský, and J. Schicho. "Graphs with Flexible Labelings." *Discrete & Computational Geometry*, 2018. online first.

[3] G. Grasegger, J. Legerský, and J. Schicho. "Graphs with Flexible Labelings Allowing Injective Realizations." Technical Report arXiv:1811.06709, arXiv, 2018.

[4] L. Henneberg. "Die graphische Statik der starren Körper." In *Encyklopädie der mathematischen Wissenschaften mit Einschluss ihrer Anwendungen*, volume IV, pages 345–434. 1903.

[5] G. Laman. "On Graphs and Rigidity of Plane Skeletal Structures." *Journal of Engineering Mathematics*, 4:331–340, 1970.

[6] H. Pollaczek-Geiringer. "Über die Gliederung ebener Fachwerke." *Zeitschrift für Angewandte Mathematik und Mechanik (ZAMM)*, 7(1):58–72, 1927.
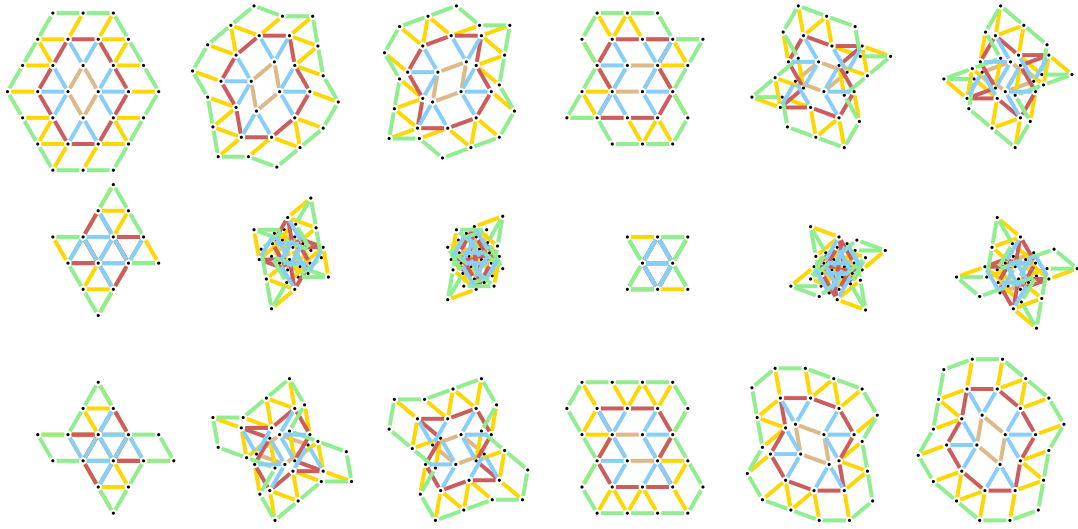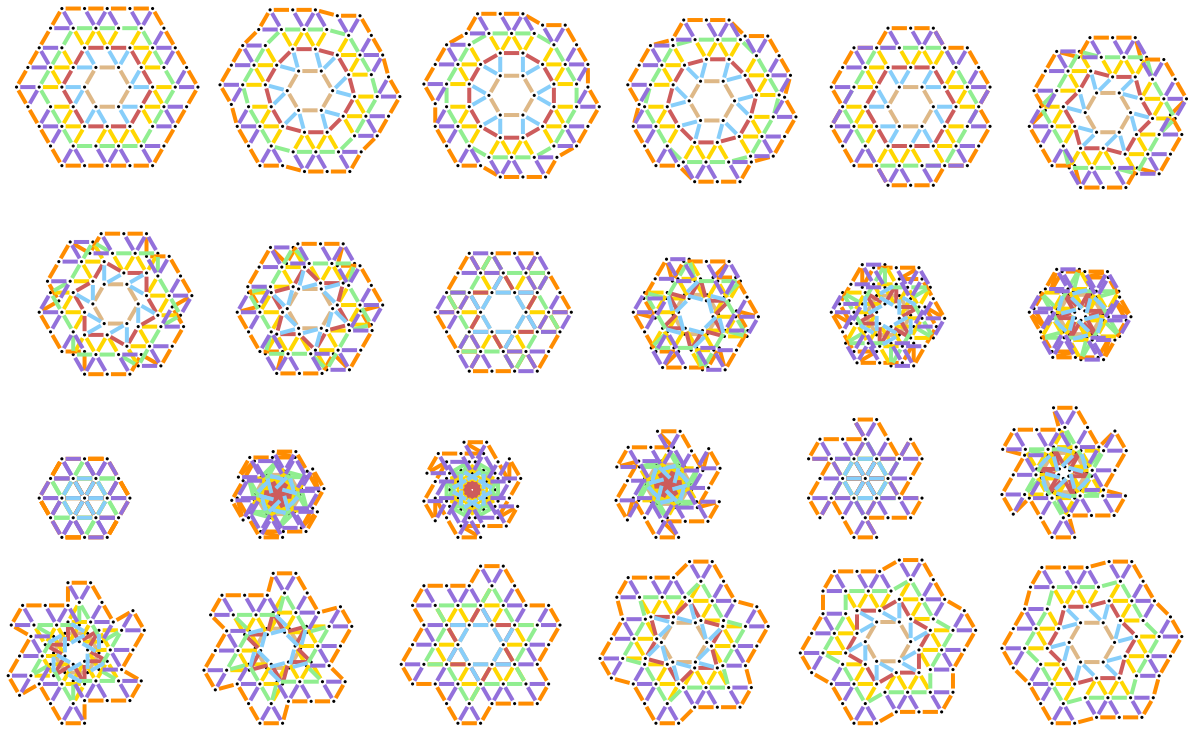
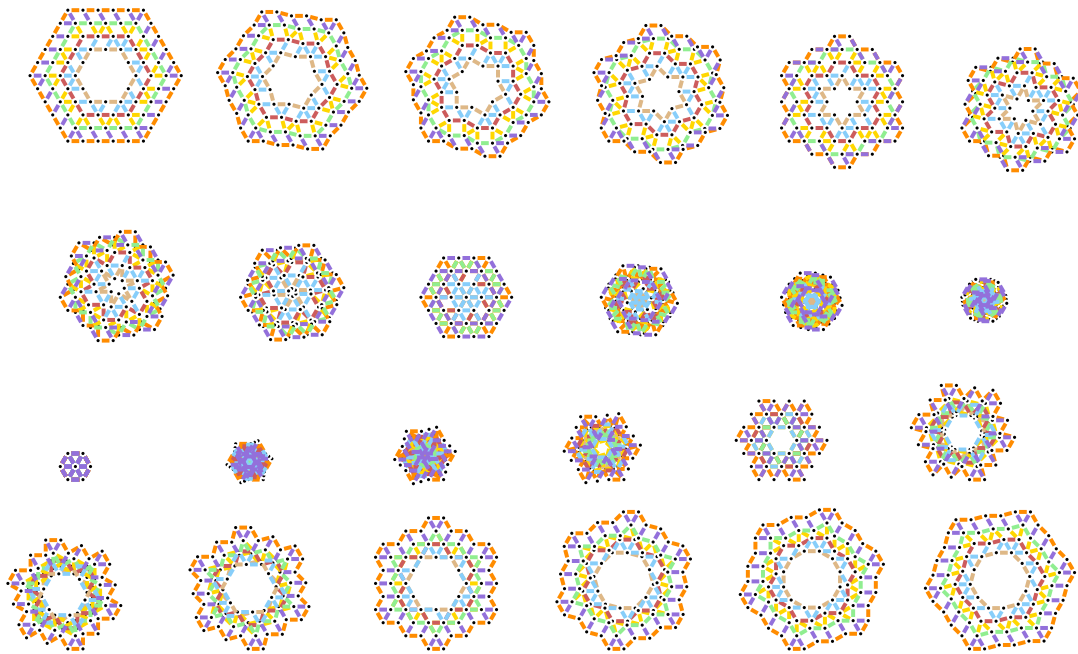**Figure 10:** *Graph with 30 vertices.*
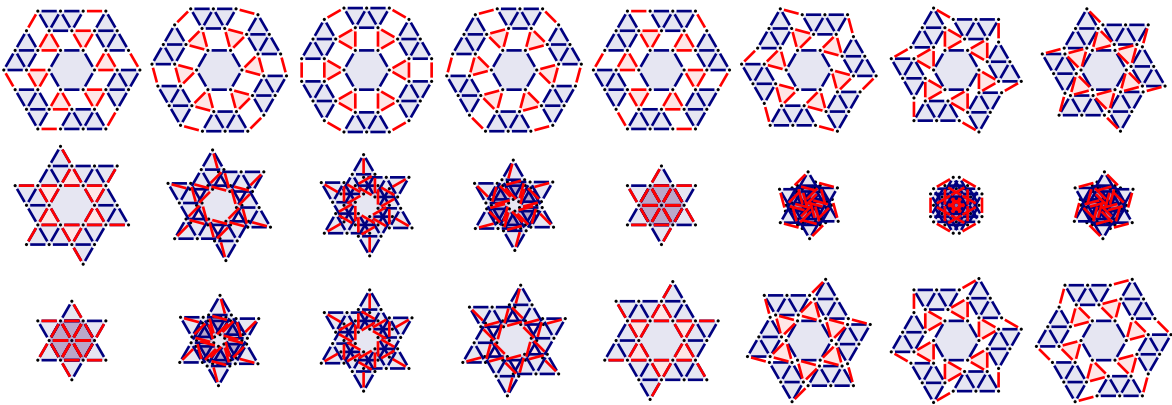
Figure 11: *Equivalence classes are highlighted by fillings.*
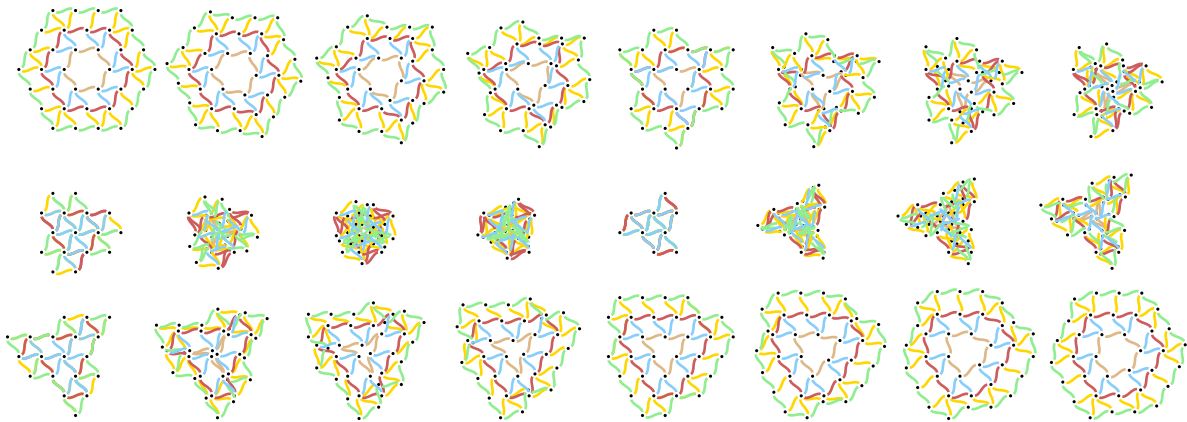
Figure 12: *Edges are curved but still their lengths do not change.*

Figure 13: *Graph with 60 vertices.*

Figure 14: *Graph with 84 vertices.*