Coloring and Counting on the Tower of Hanoi Graphs
Author(s): Danielle Arett and Suzanne Dorée
Source: Mathematics Magazine, Vol. 83, No. 3 (June 2010), pp. 200-209
Published by: Mathematical Association of America
Stable URL: http://www.jstor.org/stable/10.4169/002557010x494841
Accessed: 16-10-2015 16:49 UTC

REFERENCES
Linked references are available on JSTOR for this article:
references_tab_contents

You may need to log in to JSTOR to access the linked references.
The Tower of Hanoi graphs are intricate, highly symmetric, little-known combinatorial graphs that arise from the multipeg generalization of the well-known Tower of Hanoi puzzle. In this paper, we tour this family of graphs, exploring what we and others have shown, and what is open for further investigation. Even a quick glance at Figures 1–4 showing the first few examples (which we define more carefully within the paper) suggests patterns waiting to be discovered. We count the order, size, and degrees of vertices and show how alternate methods of counting these objects can be used to derive combinatorial identities. We describe the standard labeling of these graphs, from which we demonstrate that, although these graphs become more complex as their order increases, one measure of their complexity—the chromatic number—remains remarkably simple.
The Hanoi graphs

The graphs begin with the Tower of Hanoi puzzle. The classic version has three pegs and several disks with distinct diameters, as in Figure 5. At the beginning, all of the disks are stacked on the first peg in order by size, with the largest at the bottom. The object is to move the disks so that they are similarly stacked on the second peg. Only one disk may be moved at a time, from the top of one stack to the top of another stack (or onto an empty peg)—and, no disk may ever sit atop a smaller disk. Readers who have never tried the puzzle might wish to play one of the many available online versions.

Figure 5  The tower of Hanoi puzzle

Figure 6  Adjacent states in $H^5_4$

The puzzle was invented in 1883 by French number theorist and recreational mathematician Édouard Lucas (1842–1891). It was quickly generalized. Lucas himself explored multipeg puzzles as early as 1889. A 4-peg puzzle known as “The Reve’s Puzzle” appeared in 1908 in The Canterbury Puzzles and Other Curious Problems [3]. The problem of counting the number of steps needed to solve the multipeg puzzle (as a function of the numbers of pegs and disks) was posed in 1939 in the Monthly [17]. Lucas counted the minimum number of moves needed to solve the 3-peg puzzle, but the minimum number of moves needed to solve the 4-peg puzzle has yet to be settled. Of course, if the number of pegs exceeds the number of disks, then the puzzle is trivial, but with each added peg the corresponding graphs become more complicated. Andreas Hinz gives a more detailed history of the puzzle [4].

Associated with many puzzles and games is a model called a state graph, or configuration graph. Its vertices are the legal states, in our case the allowable configurations of disks on pegs. Two vertices are connected by an edge if a single move takes us from one state to the other. The state graph of a Tower of Hanoi puzzle with $d$ disks on $p$ pegs for $p \geq 3$ is called a generalized Tower of Hanoi graph, or just Hanoi graph, and is denoted $H^d_p$. These graphs are undirected since every move is reversible.

For example, Figure 6 shows two states in the puzzle with five disks on four pegs. We get from the first state to the second by moving the next-to-smallest (light gray) disk from the first to fourth peg. Thus the vertices corresponding to these two states are connected by an edge in the graph $H^5_4$.

To see how these graphs are built, note that for the (admittedly silly) one-disk puzzle on $p$ pegs, the state graph consists of $p$ vertices with an edge connecting each pair of vertices. That is, $H^1_p \cong K_p$, the complete graph on $p$ vertices. Another observation for those just getting to know these graphs is that the corners of the large triangle in Figure 3 correspond to states with all three disks stacked on a single peg.
For two disks, the subgraph of \( H_p^2 \) whose edges correspond to moves of the smaller disk is \( p \) disjoint copies of \( H_p^1 \cong K_p \). (Each copy of \( H_p^1 \) corresponds to a particular fixed placement of the larger disk.) To build the full graph \( H_p^2 \), we connect vertices from different components when there is a move of the larger disk between their corresponding states. For example, Figure 1 shows the graph \( H_3^2 \) built from three copies of the triangle \( H_3^1 \cong K_3 \), and Figure 2 shows the graph \( H_4^2 \) built from four copies of the kite \( H_4^1 \cong K_4 \). Using our imagination, we see \( H_5^2 \) built from five copies of the pentagram \( H_5^1 \cong K_5 \) and so on. We can more easily track this construction using the vertex labeling we present later.

In general, the \( d \)-disk graph \( H_p^d \) is built from \( p \) copies of \( H_p^{d-1} \), each corresponding to a fixed placement of the largest disk, where we connect remote vertices if there is a corresponding move of this largest disk. For example, Figure 3 shows the graph \( H_3^3 \) built from three copies of \( H_3^2 \) and Figure 4 shows the graph \( H_4^3 \) built from four copies of \( H_4^2 \).

This recursive construction suggests that the graphs are connected: that we can get from any arrangement of disks on pegs to any other in the puzzle. Though connectedness is not obvious from the puzzle itself, Hinz and Daniele Parisse prove that the Hanoi graphs are not only connected when \( p \geq 3 \), but also Hamiltonian: there exists a cycle visiting each vertex exactly once [7]. They also assert that \( H_p^d \) is \((p - 1)\)-connected: that the removal of any \( p - 2 \) vertices and their corresponding edges does not disconnect the graph.

The Hanoi graphs for the classic 3-peg puzzle were introduced in 1944 in The Mathematical Gazette [16]. They bear striking resemblance to Sierpiński’s triangles and are a special case of the Sierpiński graphs discussed by various authors [8, 9, 12, 18]. They are related to Pascal’s triangle, as discussed by David Poole [15] and Hinz [5]. As an application, Paul Cull and Ingrid Nelson discuss the 3-peg graphs’ role in perfect 1-error correcting codes [2]. The Hanoi graphs for the puzzle on more than three pegs have been studied since the 1980s, for example by Xiaowu Lu [13] and Hinz [4].

Though we are interested in the graphs, it is worth mentioning the connection to solving the puzzle. A path in a graph is a sequence of distinct vertices, each consecutive pair connected by an edge. The length of the path is the number of edges. Solving the puzzle amounts to finding a path from the starting vertex to the ending vertex, and of particular interest are paths of minimal length. In the 3-peg graphs, a minimal path follows the side of the triangle. Hinz and others have expressed hope that understanding the Hanoi graphs might lead to insight on minimal solutions of the puzzle for \( p > 3 \) pegs.

### Counting on the Hanoi graphs

A graph can be measured in many ways, often beginning with the number of vertices, number of edges, and degrees of vertices. In this section, we calculate these quantities for the Hanoi graphs. Then, we derive some combinatorial identities. These results appear (or are implicit) in the work of Sandi Klavžar, Uroš Milutinović, and Ciril Petr [10].

How many vertices does \( H_p^d \) have? Each of the \( d \) disks can be assigned to any of the \( p \) pegs. Since disks must be piled largest to smallest on each peg, each assignment produces a unique configuration. Therefore, there are \( p^d \) different configurations and, thus, \( p^d \) vertices in the graph.

How many edges does \( H_p^d \) have? For a fixed pair of pegs, we can move a disk from precisely one of those pegs to the other at every state except where both pegs are empty. Since there are \((p - 2)^d \) states with both pegs empty, there are \( p^d - (p - 2)^d \)
states where we can move a disk between this pair of pegs. Each move is counted at each state, which is to say, counted twice. Accounting for our choice of pegs as well, we find the total number of edges is

$$\frac{1}{2} \binom{p}{2} [p^d - (p - 2)^d].$$

For example, the graph $H_3^3$ shown in Figure 3 has 27 vertices and 39 edges, and the graph $H_2^4$ shown in Figure 2 has 16 vertices and 36 edges.

Alternatively, for each $1 \leq i \leq d$, we can move disk $i$ between peg A and peg B as long as none of the $i - 1$ smaller disks sit on either of these pegs. There are $\binom{p}{2}$ choices for pegs A and B, $p^{d-i}$ possible placements of the larger disks, and $(p - 2)^{i-1}$ placements of the smaller disks. Thus there are

$$\binom{p}{2} p^{d-i} (p - 2)^{i-1}$$

edges that correspond to moving disk $i$. Summing to get the total number of edges and equating with our previous count gives the identity

$$\sum_{i=1}^{d} \binom{p}{2} p^{d-i} (p - 2)^{i-1} = \frac{1}{2} \binom{p}{2} [p^d - (p - 2)^d].$$

We could have derived this by algebraic manipulation (using the factorization of $x^n - y^n$, where here $x - y = 2$), but it is more amusing when it appears from counting on Hanoi graphs.

What is the degree of each vertex? At each vertex there is one incident edge for every pair of pegs, except when both pegs are empty in the corresponding state. Thus, the degree of a vertex corresponding to a state with $k$ occupied pegs, or equivalently $k$ top disks, is

$$\binom{p}{2} - \binom{p-k}{2},$$

where the second term is understood to equal zero if $k = p - 1$ or $k = p$.

Alternatively, the only disks that move are top disks, which can move to any other peg unless that peg is occupied by a smaller top disk. Thus, counting from smallest top disk to largest, we find the degree of a vertex corresponding to a state with $k$ occupied pegs equals

$$(p - 1) + (p - 2) + \cdots + (p - k) = kp - \binom{k+1}{2} = \binom{p}{2} - \binom{p-k}{2}.$$  

Notice that the degree depends on the number of occupied pegs in the corresponding state. How many states have exactly $k$ occupied pegs? For this count we use the Stirling number of the second kind, $S(d, k)$, which equals the number of ways to partition $d$ distinguishable objects into $k$ nonempty subsets. A standard recursion to calculate $S(d, k)$ for $0 \leq k \leq d$ is

$$S(0, 0) = 1; \quad S(d, 0) = 0 \quad \text{for } d \geq 1;$$
and
\[ S(d, k) = S(d - 1, k - 1) + kS(d - 1, k), \quad \text{for } d \geq 1. \]

(To see why, note that the first summand counts the partitions where the \(d\)th element is in a singleton set.)

Thus we can sort \(d\) disks into exactly \(k\) nonempty subsets in \(S(d, k)\) ways. We can assign these subsets to \(p\) pegs in \(p(p - 1) \cdots (p - (k - 1))\) ways; we denote this \textit{falling factorial} by \((p)_k\). Since the subsequent placement of each disk onto its subset’s assigned peg is uniquely determined by size, the number of states with exactly \(k\) occupied pegs is \(S(d, k)(p)_k\).

Klavžar et al. use the Hanoi graphs to derive various combinatorial identities [10]. For example, summing over the possible number of occupied pegs and equating our two counts for the total number of vertices give the well-known Stirling identity
\[ \sum_{k=1}^{p} S(d, k)(p)_k = p^d \]
for any positive integers \(d\) and \(p\).

Similarly, we can compare the number of edges. We count \(S(d, k)(p)_k\) vertices corresponding to states with exactly \(k\) occupied pegs, each with degree \(\left(\begin{array}{c} p \\ 2 \end{array}\right) - \left(\begin{array}{c} p - k \\ 2 \end{array}\right)\). Thus the number of edges in the graph is
\[ \frac{1}{2} \sum_{k=1}^{p} S(d, k)(p)_k \left[ \left(\begin{array}{c} p \\ 2 \end{array}\right) - \left(\begin{array}{c} p - k \\ 2 \end{array}\right)\right]. \]
Equating with our previous count and simplifying give
\[ \sum_{k=1}^{p-2} S(d, k)(p)_{k+2} = p(p - 1)(p - 2)^d, \]
which might appear to be novel but, alas, after canceling \(p(p - 1)\) reduces to the same Stirling identity for \(p - 2\).

There are further enumerative uses of the Hanoi graphs. Klavžar et al. showed connections to second order Euler numbers, Lah numbers, and Catalan numbers; they suggest that there may be additional identities available [11]. Hinz et al. connect the graphs to Stern’s diatomic sequence [6].

Labeling and coloring the Hanoi graphs

It is helpful to label each vertex of the Hanoi graph in a way that lets us read off the state of the puzzle it represents. In this section, we describe the standard labeling, which leads to a natural definition of the recursive structure introduced informally earlier and is key to coloring the vertices.

It is customary to number the pegs 0, 1, 2, \ldots, \(p - 1\) and the disks 1, 2, 3, \ldots, \(d\) from smallest to largest. We say the \(i\)th disk sits on peg \(s_i\), for \(i = 1, 2, \ldots, d\), and label the vertex corresponding to this state with the string \(s_d \cdots s_2 s_1\) in this (reverse) order. Note that the labeling denotes where each disk goes; imagine placing the disks on the pegs, starting with the largest disk and working down by size.

For example, the state shown in Figure 7 corresponds to the vertex labeled 173033 in \(H_8^6\).
Figure 7 State corresponding to vertex labeled 173033 in $H_8^6$

We list the labels of its twenty-two adjacent vertices in a table.

<table>
<thead>
<tr>
<th>Disk</th>
<th>to peg 0</th>
<th>to peg 1</th>
<th>to peg 2</th>
<th>to peg 3</th>
<th>to peg 4</th>
<th>to peg 5</th>
<th>to peg 6</th>
<th>to peg 7</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>173030</td>
<td>173031</td>
<td>173032</td>
<td>173034</td>
<td>173035</td>
<td>173036</td>
<td>173037</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>173133</td>
<td>173233</td>
<td>173433</td>
<td>173533</td>
<td>173633</td>
<td>173733</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>113033</td>
<td>123033</td>
<td>143033</td>
<td>153033</td>
<td>163033</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>273033</td>
<td>473033</td>
<td>573033</td>
<td>673033</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

As another example, note that Figure 6 corresponds to the edge between vertices labeled 01302 (top) and 01332 (bottom) in $H_4^3$. Conversely, we can determine the state from its vertex label.

Notice the vertex labeled $s_d \cdots s_2 s_1$ has $k = |\{s_d, \cdots, s_2, s_1\}|$ occupied pegs. For example, the vertex labeled 173033 in $H_8^6$ has

$$k = |\{1, 7, 3, 0, 3\}| = 4$$

occupied pegs and thus degree $\binom{6}{3} - \binom{8-4}{2}$, which equals 22, as before. The reader can check the degrees in the now-labeled graphs $H_3^3$ and $H_4^2$ shown in Figures 8 and 9.

Figure 8 $H_3^3$ with vertex labels

Figure 9 $H_4^2$ with vertex labels

With this labeling we can now formally define the standard recursive construction of the graphs. We write $v \sim w$ if the vertex labeled $v$ is adjacent to the vertex labeled $w$. Any vertex of $H_p^d$ has a label of the form $av$ where $a$ is the peg number for the largest
disk and \( v \) is the label from the vertex in \( H_p^{d-1} \) corresponding to the arrangement of the other disks.

When is \( av \sim bw \) in \( H_p^d \)? There are two possibilities. If we do not move the largest disk, then \( a = b \) and, since we must move a smaller disk, \( v \sim w \) in \( H_p^{d-1} \). If we move the largest disk while the other disks remain fixed, then \( a \neq b \) but \( v = w \). In this case there cannot be any other disks on either peg \( a \) or peg \( b \) or else the largest disk could not move. Thus, in the state corresponding to \( v \), pegs \( a \) and \( b \) are empty. We abuse the notation slightly by writing \( a, b \notin v \) for short.

As an application, we derive a recursive formula for the number of edges in \( H_p^d \) for fixed \( p \), which we denote \( e_{d,p} \). An edge where we do not move the largest disk has the form \( av \sim aw \) for \( a \in \{0, 1, \ldots, p - 1\} \) and \( v \sim w \) in \( H_p^{d-1} \); thus \( H_p^d \) has \( pe_{d-1,p} \) edges of this type. An edge where we move the largest disk has the form \( av \sim bv \) for \( a, b \in \{0, 1, \ldots, p - 1\} \) and \( v \in H_p^{d-1} \) such that \( a, b \notin v \). The vertex labeled \( v \) can correspond to any of the \( (p - 2)^{d-1} \) arrangements of the \( d - 1 \) disks on the pegs other than \( a \) and \( b \). Thus \( H_p^d \) has \( \left( \frac{p}{2} \right) (p - 2)^{d-1} \) edges of this type. Therefore, \( e_{1,p} = \left( \frac{p}{2} \right) \) and for \( d \geq 2 \),

\[
e_{d,p} = p e_{d-1,p} + \left( \frac{p}{2} \right) (p - 2)^{d-1}.
\]

The reader can check that our previous count satisfies this recursion.

Thus far we have looked at known properties of the Hanoi graphs. We are now ready to prove a new result. The Hanoi graphs are complicated, but thanks to their symmetry and our convenient labeling, they can be easily colored.

For a positive integer \( c \), a graph can be \( c\)-colored if there is a way to label the vertices with the colors \( 0, 1, \ldots, c - 1 \) such that adjacent vertices are different colors. The chromatic number of a graph \( G \) is the smallest number of colors needed and is denoted \( \chi(G) \). For example, \( \chi(H^1_p) = \chi(K_p) = p \).

At any vertex of the full graph \( H^d_p \), the subgraph corresponding to moving only the smallest disk is a copy of \( H^1_p \cong K_p \). Thus \( \chi(H^d_p) \geq p \).

To see that \( p \) colors suffice, color the vertex labeled \( s_d \cdots s_2 s_1 \) by the sum of its peg numbers modulo \( p \). That is,

\[
\phi(s_d \cdots s_2 s_1) = s_d + \cdots + s_2 + s_1 \quad (\text{mod } p).
\]

To check that \( \phi \) is a \( p \)-coloring, observe that the labels of adjacent vertices differ in exactly one place, corresponding to the sole moved disk between the states.

\textbf{FIGURE 10} shows this coloring of \( H^3_p \) with white (0), light gray (1), dark gray (2), and black (3).

Alternatively, this coloring can be built recursively. Begin with \( H^1_p \) colored by its vertex labeling. For \( d \geq 2 \), given \( p \) copies of \( H_p^{d-1} \) each initially \( p \)-colored the same, place the number \( a \) in front of each vertex label in the \( a \)th copy and twist the coloring of each vertex in that copy by adding \( a \) modulo \( p \). Formally, write \( \psi(v) \) for the color assigned to the vertex labeled \( v \) in \( H_p^{d-1} \), so that the twisted coloring on \( H_p^d \) is defined by

\[
\phi(av) = \psi(v) + a \quad (\text{mod } p).
\]

The reader can now verify that each type of edge in \( H_p^d \) connects vertices of different colors and also that we obtain the same coloring as before.

Notice that, although the number of vertices and number of edges of the Hanoi graphs each grow exponentially in the number of disks, the chromatic number is independent of the number of disks.
Another way to measure a graph is by its independence number, which is the maximum number of non-adjacent vertices, usually called $\beta(G)$. In the Hanoi graphs, the $p^{d-1}$ vertices of a fixed color in a minimal coloring form an independent set and so $\beta(H^d_p) \geq p^{d-1}$. Conversely, any independent set may include at most one vertex from each copy of $K_p$ corresponding to moving only the smallest disk. As there are $p^{d-1}$ copies, $\beta(H^d_p) = p^{d-1}$.

Further investigation

While we understand much about the Hanoi graphs, there is much we still do not know. Hinz and Parisse have calculated the chromatic index (edge-coloring number) of the Hanoi graphs [8]. Any permutation of the peg numbers gives an automorphism of the graph. Recently, So Eun Park has shown that these are the only automorphisms of the graph: $\text{Aut}(H^d_p) \cong S_p$ [14]. Most graph theoretic measures of the Hanoi graphs—including the domination number, covering number, and pebbling numbers—are unknown. Some of these quantities have been calculated for the Sierpiński graphs but not the Hanoi graphs for more than three pegs [18].

We are particularly interested in the diameter: the maximum over all pairs of vertices of the minimal length of a path connecting them. The minimum number of moves needed to solve the Tower of Hanoi puzzle is bounded by the diameter of the graph and equal to the diameter in the classic 3-peg graph. The diameter of the multi-peg graphs are, in general, unknown and it is known that in some cases the diameter is larger than the minimum number of moves. Thus it is not clear whether calculating the diameter is more or less difficult than calculating the minimum number of moves needed to solve
the puzzle. Some results on the diameter of variants of the puzzle are known [1].

The 3-peg Hanoi graphs are planar: they can be drawn in the plane without any edges crossing. Hinz and Parisse [7] prove that the only planar Hanoi graphs on more than three pegs are $H^1_4$ and $H^2_4$. (We challenge the reader to draw $H^2_4$ without crossing. If you try and are stuck, consider these possibly cryptic hints: View $K_4$ as if looking at the top of a tetrahedron and do a little “cat’s cradle.” In case you are still puzzled, look for a representation of $H^2_4$ as a planar graph in the October 2010 issue of this MAGAZINE.) For any nonplanar graph, it is natural to ask about the crossing number: the minimum number of crossings needed to draw it in the plane. (Technically, a crossing involves only two edges at a time.) Alternatively we might inquire whether there are other surfaces on which the graph can be drawn without crossings; the genus of a graph is the smallest genus of such a surface. The genus is no larger than the crossing number, as one can add a bypass handle at each edge crossing, but efficiencies often lead to a smaller genus. The genera of the complete graphs are known, but the crossing numbers are not. Results on the crossing numbers of the related Sierpinski graphs are given by Klavžar and Bojan Mohar [12]. The genera and crossing numbers of nonplanar multidisk Hanoi graphs are unknown.

We offer one final direction for further investigation. Poole lists numerous variants of the puzzle [15]. For example, in “Straightline Hanoi” on three pegs, we may only move disks to and from the first peg. In “Cyclic Hanoi” the pegs are arranged in a circle and we may only move disks counterclockwise. In “Rainbow Hanoi” the disks are colored and various restrictions are placed on moves based on the color of the disks. In “Multidisk Hanoi” there are multiple copies of each disk (either distinguishable or not). Hinz claims that Lucas suggested the variation of allowing the disks to be out of order at the start—larger disks on smaller ones—subject to the usual rules later in the play. Still other variants allow a larger disk to sit on the next smallest disk, but not any smaller disks than that. To our knowledge, very little about their graphs is known.

Acknowledgment We thank Paul Cull for introducing Danielle to these graphs at his Research Experience for Undergraduates at Oregon State University in Summer 1999, Andreas Hinz for expert advice, and Matthew Richey for help with the graphics and for cheering Suzanne on.

REFERENCES


Summary  The Tower of Hanoi graphs make up a beautifully intricate and highly symmetric family of graphs that show moves in the Tower of Hanoi puzzle played on three or more pegs. Although the size and order of these graphs grow exponentially large as a function of the number of pegs, $p$, and disks, $d$ (there are $p^d$ vertices and even more edges), their chromatic number remains remarkably simple. The interplay between the puzzles and the graphs provides fertile ground for counts, alternative counts, and still more alternative counts.

DANIELLE ARETT  graduated with a double major in Mathematics and English from Augsburg College in 2000. She now works for the Hartford Life Insurance Company in Fargo, North Dakota. In her free time, Danielle enjoys writing prose, composing music, and playing piano and guitar. She first learned of the Tower of Hanoi graphs in an REU at Oregon State University in the summer of 1999.

SUZANNE DORÉE  earned her doctorate in mathematics at the University of Wisconsin. She has taught at Augsburg College since 1989 where she adores working with students—from directing undergraduate research projects in combinatorics, to helping mathematics majors develop their reasoning and speaking skills, to engaging diverse learners in the developmental algebra course she developed. For fun, Suzanne enjoys playing bridge, solving puzzles, interior design, and getting her hands dirty, literally, in the garden.