Nice Neighbors

A Brief Adventure in Mathematical Gamification

P. Christopher Staecker

Last year I came across a strange graph theory problem from digital topology. I turned it into a video game to help wrap my mind around it. It was fun to play, so I made it into a web game that other people could play. I took 3,500 unsolved math problems, made each one into a level of the game, and waited to see if people would solve my problems for me. Within two months, hundreds of people and at least one nonperson played the game, and together they solved every level.

I'll describe the mathematics behind this game and some of the surprises along the way that still have me scratching my head.

A Little Digital Topology

The game is based on a mathematical question from the fairly new field of *digital topology*, which is about topology for spaces made up of discrete pixels.

Consider Pappy, the digital character in figure 1, which we will think of as a set of 38 points. (My young daughters named him Pappy because he's purple and happy.)

Digital images like this don't obey the traditional laws of Euclidean geometry and topology. For instance, we can have two points of Pappy that are right next to each other with no point in between, which is impossible for points in Euclidean plane.

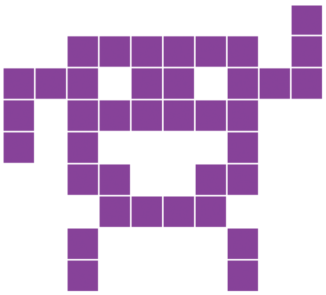

Figure 1. Pappy, a friendly digital image. Fabulous pixel art by the author.

Today is an interesting time for digital topology. The field is in its infancy, despite the fact that digital image analysis has been going on for decades in the computer industry. It's a familiar theme in the history of mathematics: Useful ideas appear in the real world before we mathematicians describe and define their theoretical foundations. (Math historian Judith Grabiner said this about calculus: "First the derivative was used, then discovered, explored and developed, and only then, defined.")

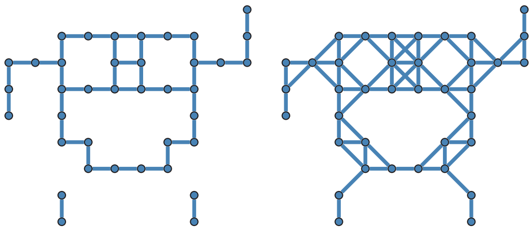

Figure 2. Two graphs modeling Pappy using 4-adjacency (left) and 8-adjacency (right).

Today there are a few approaches to digital topology that are actively being pursued. We will focus on a graph theoretical model that seems to have its origins in the work of Rosenfeld in the 1970s. The basic idea is that the topological information in a digital image is entirely expressed by which pixels are adjacent to each other. Since all we care about are the pixels and their connections, we may represent a digital image as a graph with a vertex for each pixel and an edge for each adjacency.

What do we mean by "adjacent"? There are two standard schemes: *4-adjacency*, in which diagonal neighbors are not considered adjacent, and *8-adjacency*, in which they are. Figure 2 gives the graph representations of our friend Pappy, which look like two versions of his skeleton. (Says one daughter: "That's what Pappy will look like when he's dead." The other adds, "He still looks happy.") As we can see, the topological structure of the image is quite different depending on which adjacency we use.

Reducible Images

In the summer of 2014 I was advising an REU (Research Experience for Undergraduates) project at Fairfield University with my students Jason Haarmann, Meg Fields, and Casey Peters. The idea we stumbled upon has to do with reducing a digital image by removing pixels that don't affect the topology. Just as in ordinary topology, we consider length and angle unimportant; holes and connectivity are what we care about. For example, the pixels on the ends of Pappy's hands should be removable without changing the topology.

Here's the formal idea that we settled on: When two

This content downloaded from 130.111.46.54 on Wed, 20 Apr 2016 13:50:19 UTC
All use subject to http://about.jstor.org/terms

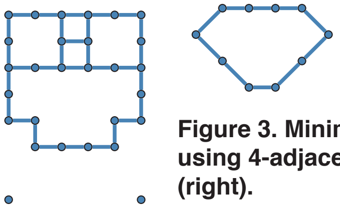

Figure 3. Minimal reductions of Pappy using 4-adjacency (left) and 8-adjacency (right).

pixels x and y are adjacent or equal, we write $x \approx y$.

A digital image X is *reducible* if there is some function $f : X \to X$ such that

1) f is not onto,
2) $x \approx f(x)$ for every pixel x, and
3) $f(x) \approx f(y)$ whenever $x \approx y$.

(In digital topology jargon, f gives a *homotopy equivalence.*) If no such f exists, then we call the image *irreducible*.

Think of f as rearranging the vertices of the graph. Then the three conditions say that

1) One vertex spot must be vacant after the move,
2) Each vertex can move only to an adjacent spot, and
3) Any adjacent vertices must still be adjacent after the move.

Our little buddy Pappy is reducible in several ways: We can chop off the ends of his arms or feet, for example. If we reduce Pappy as much as possible, we obtain the irreducible graphs in figure 3.

One major question that we tackled in our REU project was: Which graphs are reducible? Lacking any big ideas, we started small. In the end, we found the complete set of connected irreducible graphs having at most seven vertices (see figure 4). This work required proving that these graphs are irreducible and that *all* other graphs of seven or fewer vertices are reducible. (Note that we considered all graphs, not just ones that come from 4- or 8-adjacency digital images. The question "Given a graph, decide efficiently whether it is realizable as the graph of a 4- or 8-adjacency digital image" is an open problem that we didn't want to get into.)

We stopped at seven vertices because things got too complicated: We used a computer search to rule out thousands of graphs, and then we had to check 15 special cases by hand that the computer couldn't handle. For eight vertices, the number of special cases was 160, and for nine vertices it was 3,251.

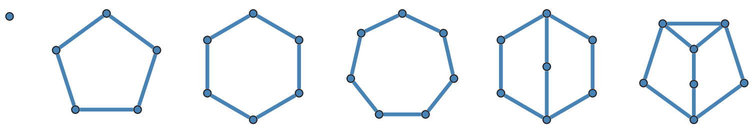

Figure 4. The catalog of all irreducible graphs up to seven vertices.

The Game

Deciding whether a graph is reducible can be hard—try figure 5 (the solution is at the end of the article). I tried to visualize the problem like one of those little board games where you have pegs that can move only between adjacent holes. But I couldn't think of a physical representation that matched the three rules for the reducing function f. Plus, I'm terrible at those peg games.

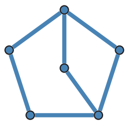

Figure 5. There is a reduction for this graph, but it had our research group stumped for a while.

A few months later, it occurred to me in the shower (where I get most of my good mathematical ideas) that this could work as a computer game. I imagined a game that keeps track of the original graph, but allows the player to move the vertices around like pegs in holes. The game would let the player move pegs in only certain ways to make sure the rearrangement satisfies the three properties.

I'm a decent programmer, and winter break was just starting, so I got to work. A few weeks later, I had a web game that gave a satisfying way to physically manipulate these graphs. In my game, the player is given a graph and can win by dragging the vertices around onto one another so that

1) One spot is vacant,
2) Each vertex ends up adjacent to where it started, and
3) Any vertices adjacent at the beginning are still adjacent at the end.

These three conditions correspond exactly to our three conditions for a reduction, so winning the game constitutes a proof that the graph is reducible. Figure 6 shows a typical game level.

Surprisingly, the game was fun. And it was possible to get better with practice, so I was developing new intuition about my math problem—though it was usually hard to articulate exactly what I was learning.

Then it occurred to me that I could code in our thousands of special cases for eight and nine vertices and crowd-source the whole thing. It seemed natural to open it up. So I came up with the deeply unsatisfying name "Nice Neighbors," and cleaned up the interface.

Marketing the game was easier than I expected. Some high-profile math tweeters mentioned it, and within a couple of weeks, I had been written up on some blogs and had all the traffic I needed.

This content downloaded from 130.111.46.54 on Wed, 20 Apr 2016 13:50:19 UTC
All use subject to http://about.jstor.org/terms

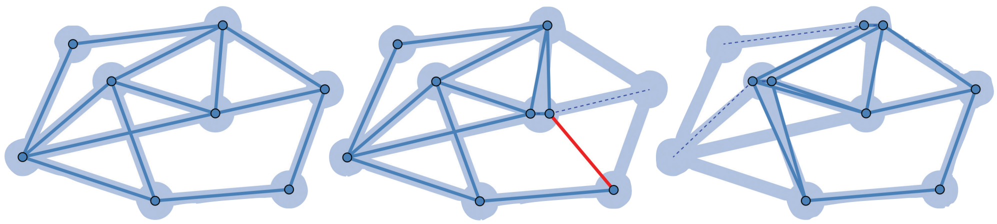

Figure 6. At left, a typical game level before any moves are made. At the middle, the player dragged a vertex to another spot. The red edge indicates that two vertices that should be adjacent are no longer adjacent, so this is not a winning position. The dotted blue line reminds the player where each vertex started. At right is a winning position for this level.

Pretty quickly I was getting feedback about the game, either directly from people who emailed me or from reading tweets and blog comments. I learned that most people won't read the instructions and that people will figure out how to cheat (especially when the game code is open). I also saw that most of the levels were solved by a small group of hard-core Nice Neighbors enthusiasts.

The biggest surprise was a particular player whom I'll call User 87. (Full name: User 8709884216. I gave each user a 10-digit identifier so I could track how many solutions were coming from each person and when.) In a day and a half, User 87 solved about one level per minute, until there were no more levels to solve. One level per minute, especially on the 9-vertex graphs, is quite a bit faster than I could consistently beat these levels, and User 87 wasn't stopping to sleep.

The only explanation is that User 87 was a computer algorithm that had bypassed the game's front-end to submit the solutions directly to my database. Had I been trolled? Was User 87 trying to humiliate me? Or maybe just trying to help? And why didn't she or he ever contact me? I still don't know. Just to be safe, I wrote my own algorithms to verify that User 87's solutions were correct. (They were.)

Within about two months, I had a complete catalog of irreducible graphs up through nine vertices. I was even able to fulfill one of my career dreams—contribute to the great Online Encyclopedia of Integer Sequences. Sequence A248571 (*oeis.org/A248571*), the number of irreducible graphs on n points (starting with $n = 1$), is now: 1, 0, 0, 0, 1, 1, 3, 28, 547. The last two terms come from the game.

But how should I feel about all this? The takeaway is: If you ask the Internet to solve a few thousand math puzzles, it probably will. But did I really learn anything, or gain new insight into my problem? Not exactly—I just have all the answers now (at least for 8- and 9-vertex graphs).

Good mathematics isn't just about finding answers, but about explaining things. There's a big difference between an answer and an explanation—these crowd-provided answers don't illuminate anything. It's satisfying for me to have the answers, but I can't escape the feeling that I still don't understand why these answers are true.

Hopefully somebody will figure it all out some day. Maybe you? Maybe me? Or maybe I'll just play my fun little game some more . . .

Further Reading

A good place to start reading about Rosenfeld-style digital topology is Rosenfeld's original paper ("Digital topology," *American Mathematical Monthly* 86 [1979] 621–630) or some later papers by Boxer (e.g., "Digitally continuous functions," *Pattern Recognition Letters* 15 [1994] 833–839).

Another important (and completely different) model of digital topology is based on the work of Khalimsky, which is in the textbook by Adams and Franzosa (*Introduction to Topology: Pure and Applied,* Pearson, 2007).

Our REU team published a paper about reducibility and homotopy equivalence (J. Haarmann, M. Murphy, C. Peters, and P. C. Staecker, "Homotopy equivalence of finite digital images," *Journal of Mathematical Imaging and Vision* 53 (2015) 288–302, arXiv eprint 1408.2584), and my algorithms can be found in "Some enumerations of digital images" (2015, arXiv eprint 1502.06236).

Nice Neighbors is still up and running for your enjoyment (*cstaecker.fairfield.edu/~cstaecker/neighbors*), as is J. Tantalo's great Planarity (*planarity.net*), which was my major design inspiration.

Chris Staecker is an associate professor in the Mathematics Department at Fairfield University, where he also teaches some computer programming. He enjoys boring movies and mechanical calculating machines.

Email: cstaecker@fairfield.edu

http://dx.doi.org/10.4169/mathhorizons.23.4

To reduce the graph in figure 5, move the five outer points counterclockwise one position, and move the center point to the top. We proved in the REU paper that any reduction of this graph must move all its points, which answered an open question posed by Boxer.

This content downloaded from 130.111.46.54 on Wed, 20 Apr 2016 13:50:19 UTC
All use subject to http://about.jstor.org/terms