A postscript to JAW17

I still remember the day I designed my first proof: April 17th, 2004. Evanston, Illinois. In my bed.

In the spring of 2004, I was a senior at Northwestern University, leading an undergraduate seminar in graph theory. I'd spent the previous year studying mathematics in Budapest, and my roommate owned a copy of Netty van Gasteren's thesis. It seemed a strange book at first, although there was definitely something to her way of thinking. From there, I found the writings of Edsger Dijkstra, and as I delved into technical note after technical note, the idea of Mathematics as Design started to sink in and transform the way I thought about mathematics.

By the time of the seminar, I could follow most of the EWDs pretty well, and could confidently speak about the merits of the "Dutch style" in mathematics. And I could critique the work of others, show where a mess had been made of things. But I had yet to ever design a proof or program of my own.

In one of the first lectures, I needed to present a proof of the fact that a graph is 2-colorable if and only if it has no odd cycles. I'd seen many such proofs, but all of them dissatisfied me now: Too many case distinctions, too many details swept under the rug, or rabbits pulled out of the proverbial hat. I wanted to do better, so I lay down in my bed and began to think. [If you don't know anything about graph theory and want to follow what comes next, flip to the Appendix.]

I already knew a pretty nice proof that 2-colorable graphs have no odd cycles: Given a 2-colorable graph, properly color the nodes red and blue. Since all edges are mixed, nodes alternate color along a path. Now, an odd cycle is formed by an even path, followed by an edge connecting the first and last nodes. But in an even path, the first and last nodes have the same color, hence there can be no edge connecting them. Hence a 2-colorable graph has no odd cycles.

The bigger question was how to prove that a graph with no odd cycles is 2-colorable. Most proofs start by assuming ("without loss of generality") that the graph is connected, then color one of the nodes arbitrarily, then proceed to color the rest of the nodes. I didn't want to make case distinctions and break symmetries.

So I started to think the Dutch way. What are we trying to accomplish? To color all the nodes so that each edge is mixed. So even though we're coloring the nodes, the crucial condition is about edges. Our algorithm, rather than proceeding by coloring nodes, should proceed by checking edges!

We can initialize the algorithm by removing all the edges, and coloring the nodes however we like. Now, trivially, "all" edges are mixed! To proceed, we have to replace edges one by one, at each step ensuring that all edges remain mixed. I remember the thrill of wondering if this method would work, and if so, how.

Suppose we have done our job so far, and that all replaced edges are mixed. Now we need to replace another edge xy. Can we do it? I sought to show that x and y had different colors, and I knew somewhere I had to use the fact that there are no odd cycles. Since xy is an edge, any path connecting x and y is odd, otherwise the graph would have an odd cycle. But if x and y are connected by an odd path, then since colors alternate along a path, x and y indeed have different colors, so xy can be replaced!

Was I done? I thought I was. After all, I'd used the given information. But thinking through some concrete examples showed me that something was wrong. It can't be the case that any time we need to replace edge xy, x and y have different colors.

I quickly realized that if x and y are connected, then they are indeed connected by an odd path, and they indeed have different colors. However, what if they are not connected? We might get lucky and they just happen to have different colors. Then we can replace xy.

But what if x and y are not connected, and they have the same color? What do we do? Lying on the bed, I imagined a red x, part of its connected component, and a red y, part of a different connected component. With that picture in my mind, the solution presented itself almost immediately: simply swap colors in one the two components!

The whole process had taken no more than a minute, two at most. I had followed the principles of the Dutch school and had been rewarded with a solution which was simple and elegant.

(I later streamlined the argument: When we replace xy, first check if x and y have different colors already. If they do, we can replace xy without any changes. If x and y have the same color, then they cannot be connected, otherwise they would be connected by an even path, and this path together with xy would form an odd cycle. Hence in this case x and y are in different connected components and we can swap colors in one of them, and then replace xy.)

(Also, my proof does not work for infinite graphs.)

I was breathless. I had designed this little gem seemingly with no effort whatsoever. The great invention of the color swap, far from being a rabbit pulled out of a hat, was dictated by the very needs of the proof itself. For the first time, I understood how invention could be driven not by divine inspiration, but by simply paying attention.

Further, my proof had no needless nomenclature, no subscripts. No arbitrary destruction of symmetry by picking a special node to color first. As a result, my proof didn't get bogged down in the end with case analysis. And whereas most proofs start by assuming the graph is connected, an immediate case distinction with no apparent reason, my algorithm thrives on the disconnectedness of the graph!

Once I'd designed this first proof, I felt free. It wasn't long before ideas for JAWs began piling up. I got a Fulbright grant to study in the Netherlands with Wim Feijen, Tom Verhoeff, Apurva Mehta, and others. "Mathmeth" was born, and the direction of my life seemed to have changed forever.

Of course, that's not what happened. I was never able to continue studying calculational mathematics, despite applying for many grants, and I eventually moved to New York City to pursue a career in classical music. A few years later, I met my wife, and now I'm a stay-at-home dad to two wonderful young boys.

Nevertheless, that beautiful, horizontal experience in 2004 changed my very being forever.

It's worth pointing out that I had trouble writing up this proof for many years. It took almost two years to write up my design as JAW17 , but that note is a very different beast, one I've never been completely satisfied with. In that note, I aimed to motivate every aspect of the design, and as a result, I never quite captured elegance and thrill of my first design.

In between April 17th, 2004 and JAW17 , I had many false starts. I recorded myself talking through the proof. I presented it to colleagues. But I couldn't get anywhere writing it down. Eventually I learned why: My proof had made some hidden assumptions, and my sensitivity at that time to clarity was so extreme, I simply couldn't let myself take a step without knowing exactly where I was going and why. (Unbeknownst to me, the traditional proofs had the same hidden assumptions, but they were so hand-wavy, I hadn't even recognized it!)

As I approach the 16-year anniversary of my first design, I feel compelled by the distance in time and life circumstances to recreate some of the feelings of that magical day. I can only hope that my children, my students, my readers, can experience something equally profound, if they haven't already!

Appendix

The following definitions are not necessarily meant to be mathematically useful. They're simply meant to give an inexperienced reader some idea of what's going on.

A ‘graph’ is a collection of ‘nodes’, where every pair of nodes is either ‘joined’ or not by an ‘edge’. If nodes x and y are joined, we call the edge joining them xy.

A ‘walk’ through a graph is what you take starting at a node, stepping via an edge to another node, and so on. A ‘path’ is a walk with no repeated nodes, and a ‘cycle’ is a path followed by one more step from the last node to the first. A walk, path, or cycle is called ‘odd’ or ‘even’ according to whether the number of edges (steps) in it is odd or even.

Two nodes are ‘connected’ if there is a walk from one to the other. A graph is ‘connected’ if all its nodes are connected. A ‘connected component’ of a graph is a collection of nodes that are connected to each other but no others.

We ‘color’ a graph by assigning each node a color. In a colored graph, we say that an edge is ‘mixed’ if the nodes it joins have different colors. If every edge is mixed, we say that the graph has been ‘properly colored’. A graph that can be properly colored with two colors is ‘2-colorable’.

(End of Appendix.)

NYC, March 8th 2020

Jeremy Weissmann
jeremy@mathmeth.com