Spiderwire: Visualizing Calculus

Summer 2014, NSERC USRA Report *Emily Tyhurst*

The Spiderwire project, formerly known as the visualizing calculus project, is an online math education tool for the understanding of logical connections and implications. Spiderwire takes the form of a graph, where nodes representing theorems are connected by edges which represent their proofs.

Introduction and Motivation

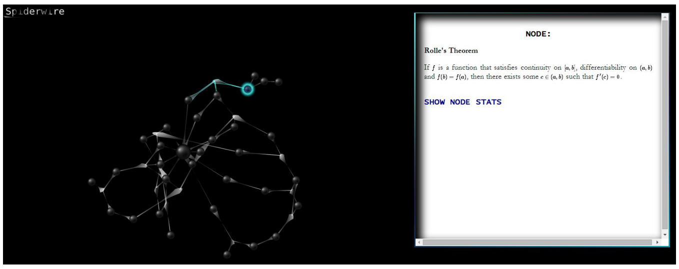
The idea of mathematics as a network of connected ideas, with every result owing some debt to fundamental constructions of our number system is one that is familiar to many advanced mathematicians, but perhaps not familiar to the first year, second year, or even third year student (there is some expectation that either MATH 320 or MATH 220 should give one an idea of how to take mathematics to its foundations). This project aimed to create a compelling visualization for these tenuous connections, giving both the student and the academic a greater understanding of how mathematics is developed, and how high-level results may be traced back to elementary results.

The secondary goal of the project was to improve access to proofs and logic to students in first year calculus, especially within courses where curriculum is often too stretched to really give students exposure beyond applications (which tend to be more essential to students not going on in mathematics). The content within *Spiderwire* spans that of a first-year differential calculus course, with some overlap into integral calculus which was necessary to demonstrate results rigorously.

The project is currently on the UBC Math Department website at **http://l14-fsl.math.ubc.ca** for students and professors to refer to as a resource.

Functionality

The main functionality involves interacting with a web of nodes and links. Clicking on a node will produce a theorem and its result, given certain conditions. Clicking the edge before a node will give that theorem's proof. Certain theorems are dependent on more than one theorem, so their edges join up to form super-edges, all of which are revealed on click. The proof and theorem descriptions include vocabulary words tagged for students who are not yet accustomed to the rigorous terminology within Calculus.



NODE:

Rolle's Theorem

If f is a function that satisfies continuity on $[a, b]$, differentiability on (a, b) and $f(b) = f(a)$, then there exists some $c \in (a, b)$ such that $f'(c) = 0$.

Figure 1: *Selecting a node within Spiderwire.*

In addition, the larger nodes (currently only the Completeness Property) contain more than one result or theorem that are connected by "equivalence". This relationship requires that any given result (scoped under the larger node) can be said to imply the others in a cycle. In the case of the Completeness Property, this means that given any of the results: Least Upper Bound Property, Monotone Convergence Theorem, Nested Intervals Theorem, Bolzano-Weierstrass Theorem, Cauchy Sequence Convergence, all the other results follow. Visually, these are scoped as "sub-nodes" and "sub-links" that are viewed in a smaller graph when the large node is selected.

Subject, Method and Course Tags

As an educational component, all objects in *Spiderwire* are tagged with subject tags and course tags. The "subject" tags can be viewed under the "Stats" of a certain node or link. This aids students who are new to mathematics as an academic discipline get familiar with what types of disciplines of mathematics exist, and typical results within that discipline. There is exists an option within the "Browse" component of the graph interface to highlight all proofs and results tagged with a subject (or subjects).

In addition, proofs are given "method" tags which describe canonical methods within mathematical proof (ie. Proof by Bisection, *reducio ad absurdum*, δ-ε Limit Defintion, etc.). This allows students to develop familiarity with the tools of mathematical proof, and where they are most appropriate. A similar highlighting functions shows proofs tagged with a given method (or methods).

The "course" tags currently exists only within the database of *Spiderwire*. They seemed to be a relatively meaningless educational tool, however, in the event that the database does expand to many courses, it might be interesting for the user to see where two different courses intersect. It could also be an economical way to make decisions regarding which nodes to omit, should the graph contain over the limit of viewable objects. For these reasons the course tags have remained as a development possibility.

Creating a New Weave

In addition to viewing most of the typical differential-calculus content, users have the option of viewing nodes and links by certain criterion. The most useful of these features is the Ancestor/Children weave. Users can choose to see the ancestors (all theorems used to prove) of a specific node, or the children (all theorems proved by or with that node's contribution to the proof) of a specific node. This reveals information about how fundamental or complex a result is.

In addition, there is a possibility of using a string "search function" to generate new graphs. Depending on the quality of the search criteria, this can reveal interesting information. The idea of generating graphs by subject, course, or method was investigated, however due to the occasionally arbitrary nature of identifying results as belonging to specific subjects, this option was not encouraged. It remains possible to generate these odd-weaves based on words, subjects, or method parameters, but they reveal less interesting information about the connectedness of the results.

Design and Programming

Spiderwire was designed using the Javascript library d3.js, which assigns data from a Django (Python) webserver to SVG objects. The graphing algorithm is force-directed, meaning that the nodes have a certain repulsion towards each other coupled with a certain attraction towards the center, which allows for a centralized but well-spread graph. The links act as springs, bringing the nodes back into view as they are dragged an rearranged.

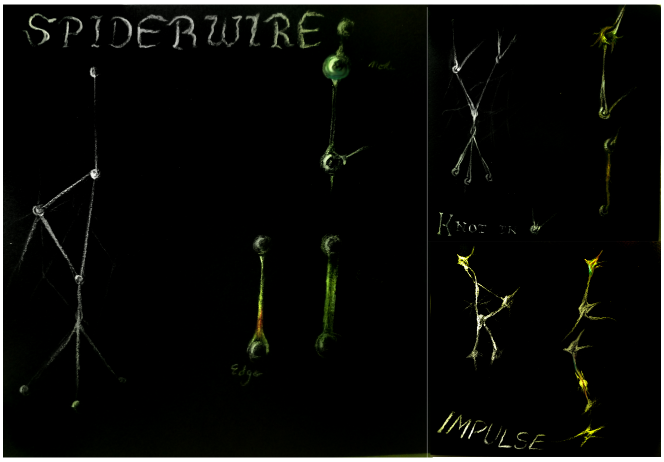


Figure 2: *Original design proposals for the Visualizing Calculus Project.*

The overall design of the system was inspired by that of a spider's web, and is meant to equally celebrate organic creativity within mathematics as well as the stunning technological advances that mathematics continues to facilitate.

Content and Questions Related to the Teaching of Calculus

The content of the Calculus is unique in that it depends so centrally on the idea of a Completeness Property that all of the fundamental results can be thought of as implicitly dependent on this axiom (for the purposes of first year calculus, we did not formulate any constructions of $\mathbb{R}$).

The Completeness Property is normally axiomatically stated as the Least Upper Bound Property- which can seem both arbitrarily simple and strangely difficult to grasp. By establishing it as equal with more advanced theorems which imply Completeness (Bolzano-Weierstrass, Cauchy Sequences, etc.), the student is given a richer understanding of its significance and utility.

Interestingly enough, most of the results whose proofs one would expect to see in first-year Calculus were two or three tiers out from the Completeness Property- the fundamental laws of limits (according to Cauchy's definition) had the closest "implicit" links to the Completeness Property (Within *Spiderwire*, a theorem dependency was cited as "implicit" if the conditions *could* depend on an axiom, but not necessarily. For example, the statement "$\lim_{x\to a} f(x) = L$" could be meaningful in $\mathbb{Q}$ if $L \in \mathbb{Q}$. But if we want the extended results that $L \in \mathbb{R}$, then the Completeness Property is necessary).

The most complex node (the one with the most ancestors) was the Power Rule for Differentiation- in order to give a sufficiently rigorous, yet understandable definition, some results within integral calculus were used. This illustrates the connectedness of results within the field of calculus (and mathematics in general).

Ideally, many instructors of mathematics would appreciate a bit more analysis and integral calculus to increase the depth of a simple differential calculus course. *Spiderwire* will provide a forum for these ideas that will hopefully provide opportunities for discussion and learning.

Future Directions

Spiderwire was always intended to go beyond the limitations of a first year Calculus course, even beyond Calculus itself. Now that the groundwork for the project is laid, we hope as it gains popularity, there will be interest in adding entirely new course material to the database, and perhaps understanding how two different fields may link themselves. At present moment, the admin interface is not particularly intuitive, nor is the graph interface well-equipped to deal with more than about 40 nodes (before the sizing tuning should probably be adjusted). A "user manual" has been prepared with the hope that this project may be further developed at a later time.

The project will undergo testing for the upcoming year. A survey has been developed to gauge reactions to the project and where the project may be improved in the subsequent versions.