Research Statement

Nicholas Connolly

Abstract

My doctoral research is focused on the tabulation and classification of the knot theoretic structures known as tangles. Tangles can be understood most generally as the building blocks of mathematical knots. The enumeration of knots has been the quintessential goal of knot theory since its inception, leading some to describe tangle tabulation as "the most important missing infrastructure project" in the field [2]. My doctoral research has two principal goals. First, I am addressing this deficiency by creating an exhaustive database of 2-string tangles and their properties. Second, I am developing a computational algorithm to classify the structure of a tangle diagram through a decomposition into subtangles. This research statement provides a brief overview of my achievements on these projects.

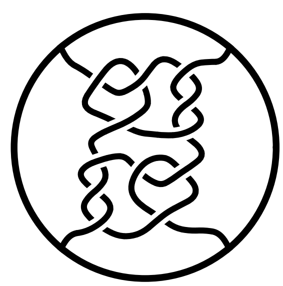
In addition to my research as a doctoral candidate in mathematics, I have also participated in multiple non-academic research internships during the summers of 2019 and 2020. These interdisciplinary internships have greatly complemented my studies in pure mathematics with applied experience in data science and machine learning. Brief descriptions of these projects are included following the discussion of my dissertation research.

Introduction

Background

A tangle is a three dimensional object consisting of a ball embedded with multiple entwined strings (Figure 1). The well known knot theorist J. H. Conway first introduced tangles as a way to tabulate knots [9]. He showed that it was possible to describe any knot with up to 11 crossings in terms of one of eight basic planar graphs, work that was later improved by Caudron [4]. Tabulation of knots and links is one of the classical goals of knot theory, with many mathematicians contributing to this endeavor over the years [14, 13, 17, 22, 19, 30].

20 24 22 12 16 | 14 6 10 8 2 18 4



$\left(\frac{2}{3}+\frac{1}{3}\right) * \left(\frac{1}{3}+\frac{2}{3}\right)$

Figure 1: Example of a tangle diagram, generated using KnotPlot [27], and two different diagram notations: the Dowker code (above) and an algebraic construction formula (below).

By contrast, tabulation of tangles has been less thoroughly researched, although progress has been made in special cases [9, 20, 25] and for small crossing numbers when allowing the boundary to move [29, 18]. Depending on the application, tangles can be studied either allowing the endpoints to move on the boundary sphere or requiring them to be fixed. My dissertation research is concerned with the study of 2-string tangles of both types. Motivated by Conway's work, I am continuing the project of tangle tabulation using a graph theoretic approach to decompose tangle diagrams into subtangles. By enumerating these decompositions, I can generate an exhaustive list of tangles up to a specified crossing number.

While tangles are three dimensional objects, they are primarily studied using two dimensional tangle diagrams. A diagram may be uniquely encoded as a sequence of numbers referred to as a diagram notation, such as the Dowker code [14] (Figure 1). There are many such notations, but each allows the topology of a tangle diagram to be studied using combinatorics and computational search algorithms. I have developed my own *subtangle planar diagram* notation to decompose a diagram into subtangles based on the idea that large tangles with a complicated structure can be described using small tangles with a simple structure. My notation is a generalization of Conway notation and a notation introduced by Ewing and Millett [16], which describes a similar decomposition of a knot diagram into crossings. I have successfully used my new notation to classify and tabulate tangle diagrams and I am now focused on consolidating my results in a web-accessible database [6].

Broader Impact

Within the field of topology in general and knot theory in particular, the ability to demonstrate major results using examples is critical to understanding the subject. As any textbook in the discipline shows [26, 1, 10], figures, diagrams, and tables of knots are ubiquitous among publications in knot theory. As the building blocks of knots, topologist Dror Bar-Natan has proclaimed the importance of tangle tabulation on the basis that tangles are precisely what the knot theorist must study [2]. My contribution to this growing body of work is an organized database of tangle examples. Beyond pure knot theory, tangles also play a prominent role in mathematical biology, where they have been used to model protein folding and to solve tangle equations modeling DNA bound by protein [15, 11, 3, 12]. By developing this database, I am creating a resource that is available to the broader scientific community.

Overview of Doctoral Research

To expand upon my research objectives and accomplishments, the discussion of my doctoral research is divided into the following five sections.

1. **Families of Tangles:** Broadly speaking, 2-string tangles can be divided into families based on their structure: integer, rational, Montesinos, algebraic, and non-algebraic. I begin with a brief description of these families.

2. **Classification Algorithm:** Using graph theory, I have developed a *subtangle planar diagram* notation that describes a tangle diagram in terms of a decomposition into subtangles, a notation which makes it easy to identify the family of a diagram. Furthermore,

I have developed and implemented an algorithm in C/C++ to compute this notation for a given tangle diagram. This section outlines the basic concept of my algorithm.

3. **Tangle Tabulation:** This section discusses a computational method for tabulating a list of tangles up to fixed crossing number. The method is based on exhaustively generating diagram notations and eliminating known redundancies.

4. **Database Development:** The combined result of my classification and tabulation programs, I have created a working prototype of a web-accessible database for 2-string tangles using SQL, HTML, and PHP (web address included in references [6]). The prototype provides an early illustration of my goals for this resource for some special families of tangles.

5. **Future Work:** While the 2-strings tangles database is the culmination of my graduate research, I have several ideas for new directions to expand upon as a postdoctoral researcher. I conclude with a discussion of some of these plans.

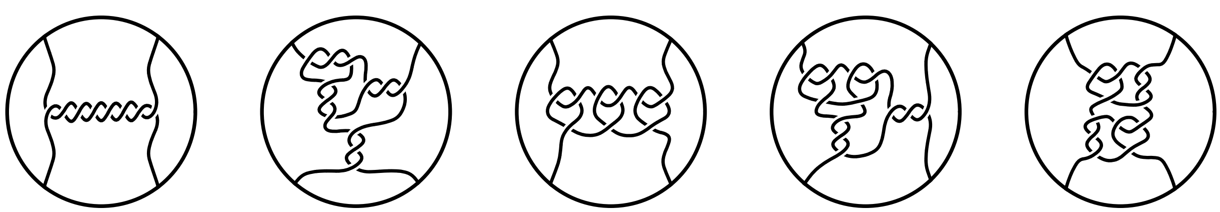
Doctoral Research

Families of 2-String Tangles

There are six families of 2-string tangles, with examples for the first five families shown in Figure 2. **Integer** tangles are the simplest family to understand, and consist purely of a horizontal sequence of twists. Figure 2a shows an example of an 8-crossing integer tangle, which is denoted by the fraction $\frac{8}{1}$. More generally, a horizontal or vertical sequence of twists is referred to as a horizontal or vertical tangle. These tangles are denoted by the fractions $\frac{n}{1}$ and $\frac{1}{n}$, respectively, where $|n|$ is the number of crossings and the sign of n matches the slope of the over-strand.

Figures 3a and 3b show how two tangles can be stacked together like blocks to build more complicated diagrams. The operation of joining two tangles horizontally is called *tangle sum*,

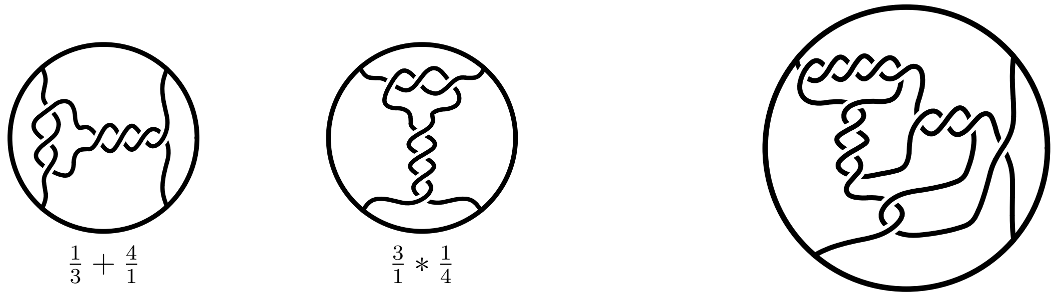
(a) Integer **(b)** Rational **(c)** Montesinos **(d)** Gen. Montesinos **(e)** Algebraic



$\frac{8}{1}$ $\frac{33}{109}$ $\frac{2}{3}+\frac{2}{3}+\frac{2}{3}$ $(\frac{2}{3}+\frac{2}{3})\circ(3,3)$ $(\frac{2}{3}+\frac{1}{3})*(\frac{1}{3}+\frac{2}{3})$

Figure 2: Examples of tangle diagrams for several families of 2-string tangles, organized by increasing generalizability. The hierarchy of those shown is: {Integer} ⊆ {Rational} ⊆ {Montesinos} ⊆ {Generalized Montesinos} ⊆ {Algebraic}. Since each of these examples is an algebraic tangle, the tangles above can be denoted with a compact algebraic construction formula.

(a) Tangle Sum **(b)** Tangle Product **(c)** Rational tangle with twist vector $(5, 4, 3, 2, 1)$



$\frac{1}{3} + \frac{4}{1}$ $\frac{3}{1} * \frac{1}{4}$ $\frac{225}{157}$

Figure 3: Tangles can be stacked together to build larger tangles, operations referred to as tangle sum (horizontal join) and tangle product (vertical join). Rational tangles are built using sums and products of horizontal and vertical tangles. The construction of a rational tangle is summarized by a twist vector $(n_1, \cdots, n_k)$ that defines a continued fraction $n_k + \frac{1}{n_{k-1} + \cdots + \frac{1}{n_1}} = \frac{p}{q}$. This fraction defines a bijection between rational tangles and the extended rational numbers [9, 20].

while joining two tangles vertically is called *tangle product*. A tangle is **rational** if it can be constructed using sums with horizontal tangles and products with vertical tangles. A special feature of rational tangles is that all twists can be shuffled to the right and bottom sides of the diagram, giving a preferred construction like the ones shown in Figures 2b and 3c. From this preferred construction, a rational tangle can be described by a sequence of numbers $(n_1, \cdots, n_k)$ called a *twist vector* which denotes alternating right and bottom twists. As with integer tangles, twists may be either positive or negative depending on the slope of the over-strand, but a rational tangle diagram will not be minimal unless all twists have the same sign. The twist vector defines a continued fraction that simplifies to a unique extended rational number $\frac{p}{q}$ that is used to denote the rational tangle (see Figure 3).

Rational tangles have been widely studied because of their particularly well-behaved structure. **Algebraic** tangles are a generalization of rational tangles which have not yet been classified uniquely. A tangle is algebraic if it can be constructed using any combination of sums and products of rational tangles, yielding a diagram described by a compact *construction formula* like the ones shown in Figure 2. While a preferred diagram for generic algebraic tangles is not yet known, such a description does exist for some special cases [28]. A **Montesinos** tangle is an algebraic tangle constructed using only tangle sums, as shown in Figure 2c. A generalized Montesinos tangle consists of a Montesinos tangle that also has external right and bottom twists. Unlike generic algebraic tangles, generalized Montesinos tangles do admit a preferred construction [24, 25].

If one allows the boundary of a tangle to move, the endpoints of the strands on the surface of the ball can unwind. Under this definition of tangle, all generalized Montesinos tangles are equivalent to a Montesinos tangle. In particular, all rational tangles are equivalent to a 0-crossing tangle. I am interested in the study of tangles both with and without a fixed boundary. For fixed boundary tangles, this external twisting can be described using the same twist vector notation used for rational tangles. A generalized Montesinos tangle is often denoted with a core Montesinos tangle and a twist vector (Figure 2d).

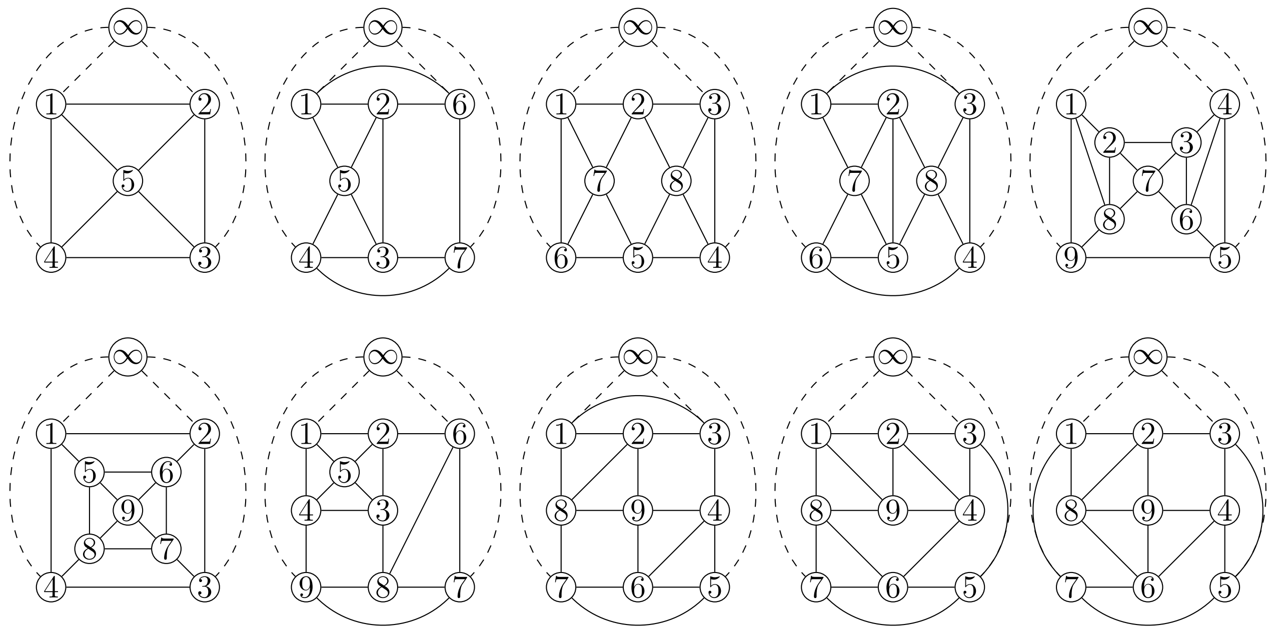


Figure 4: The first 10 constellation graphs showing possible non-algebraic tangle diagram configurations. These graphs can be used to build diagrams for non-algebraic tangles by replacing the vertices with algebraic subtangles (Figure 5c). The special vertex marked ∞ denotes the four endpoints of a tangle, where we imagine collapsing these to a single point. In combination with tangle sums and products, all non-algebraic tangle diagrams with at most 9 crossings can be described using one of these constellations. Constellations are closely related to Conway's graphs [9].

The five tangles shown in Figure 2 are all examples of algebraic tangles. A **non-algebraic** tangle is one that cannot be constructed using only sums and products of rational tangles, which makes them difficult to describe in general. Even though they cannot be constructed using only tangle algebra, non-algebraic tangle diagrams can still be decomposed into certain configurations of algebraic subtangles. By studying how to decompose tangle diagrams into subtangles, I have discovered that configurations for non-algebraic diagrams can be modeled using certain marked 4-valent planar graphs, structures which I refer to as *constellations* in my research. By comparison with existing tables of planar graphs [23], I have identified up to isomorphism 10 distinct constellations (Figure 4) which can be used to classify all non-algebraic tangle diagrams with at most 9 crossings. For example, the non-algebraic tangle diagram of Figure 5c can be decomposed into into five algebraic subtangles using the first constellation of Figure 4.

Constellations are a generalization of the planar graphs Conway used to tabulate knots [9], and Conway's graphs can be recovered from this list of constellations by closing tangle diagrams into knots. The Conway notation for a knot describes how tangles can be substituted for the vertices in a graph to build a diagram. Using the same idea, I have extended this to tangles using a similar constellation notation (Figure 5b). This graph theoretic description of tangle diagrams plays a major role in my research.

(a) Subtangle Planar Diagram Code

(2/3)	0 NW	2 NW	5 NW	4 SE	(P : 0′)
(3/2) ∗ (2/1)	1 NE	0 NE	3 SW	5 NE	(P : 0)
(4/1)	0 SE	4 NW	5 SE	2 SE	(P : 0)
(1/3)	3 NE	0 SW	1 SW	5 SW	(P : 1′)
(3/7)	1 SE	2 SW	3 SE	4 SW	(P : 1)

(b) Constellation Notation

$$C_{5+1}\Big(\big(\tfrac{2}{3}\big), \big(\tfrac{3}{2} * \tfrac{2}{1}\big), \big(\tfrac{4}{1}\big), \big(\tfrac{1}{3}\big), \big(\tfrac{3}{7}\big) \Big)$$

(c) Non-Algebraic Diagram Decomposition

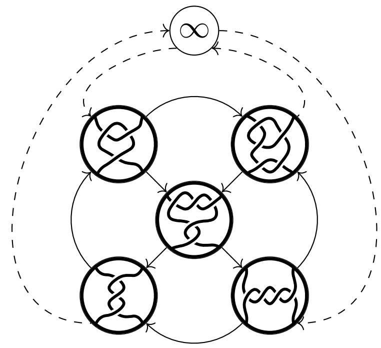


Figure 5: Example of a non-algebraic tangle diagram decomposed into algebraic subtangles. The configuration of subtangles shown in 5c matches the first constellation graph of Figure 4. In the corresponding subtangle planar diagram notation (5a), each row denotes an algebraic subtangle and its connections to other subtangles. (See [6] for additional details about this notation). The connection information describes the adjacency list of the graph for this subtangle decomposition. In combination with tangle sums and products, constellation notation (5b) is a compact way to represent a non-algebraic diagram by slotting in subtangles for the vertices in a constellation graph.

Classification Algorithm

The construction of an algebraic tangle diagram is explicitly described by a sequence of sums and products of rational numbers, such as those shown in Figure 2, which corresponds to a certain configuration of rational subtangles. Hence, one way to distinguish algebraic and non-algebraic diagrams is by finding and analyzing such a subtangle decomposition. The original diagram is algebraic if and only if this configuration can be expressed with tangle sums and tangle products, which defines the corresponding sequence of rational numbers. For algebraic diagrams, it is easy to identify the tangle family based on this sequence. Therefore, a tangle diagram can be classified by computing a subtangle decomposition.

By generalizing the Ewing and Millett notation [16], which decomposes a diagram into crossings, I have developed my own *subtangle planar diagram* notation which describes an explicit configuration of subtangles in terms of the adjacency list for a graph. This notation is not unique in general since the same diagram can be modeled using a different combination and configuration of subtangles. However, this notation will be unique if decomposing a diagram into maximal algebraic subtangles. Any non-maximal subtangle decomposition can be refined by joining together neighboring subtangles that share algebraic connections (Figures 3a and 3b), until a decomposition into maximal algebraic subtangles is obtained.

A major result in my research is the implementation of an algorithm in C/C++ [5] to find this decomposition by computing the corresponding subtangle planar diagram code. Beginning from the Ewing-Millet notation, my algorithm converts this into a decomposition of 1-crossing subtangles. It then iteratively refines this decomposition by searching for subtangles that can be joined together with sums or products, and terminates when no remaining subtangles share an algebraic connection. The result is either the construction formula for an algebraic tangle, or a list of connected algebraic subtangles (Figure 5).

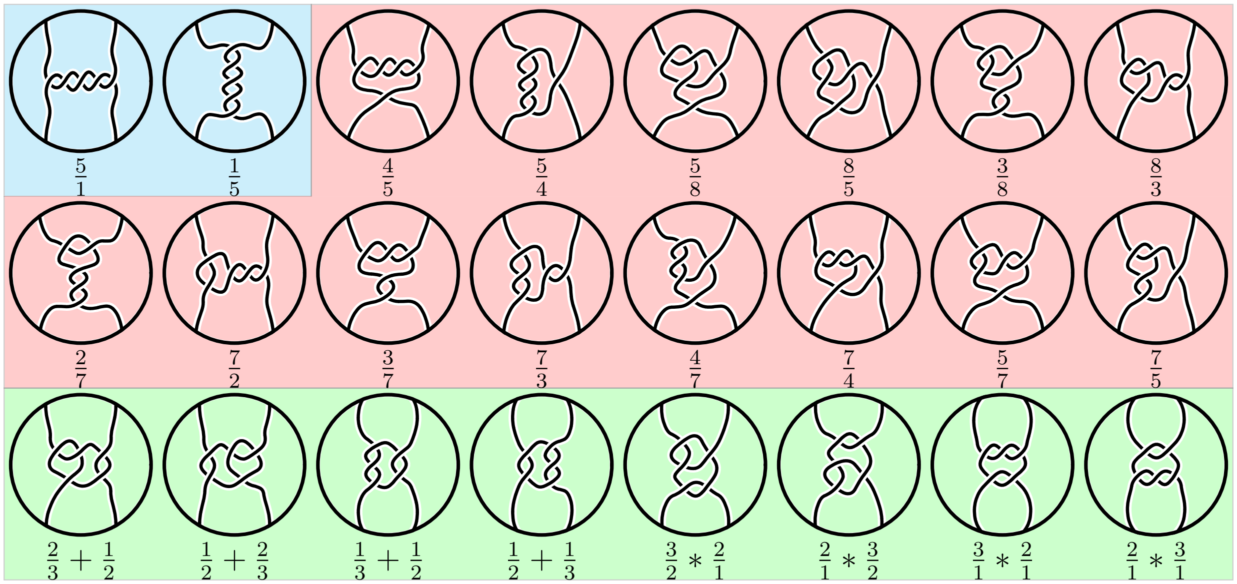


Figure 6: Table of 2-string tangles with 5 crossings, excluding mirror images. Blue denotes horizontal and vertical tangles, red denotes rational tangles, and green denotes Montesinos tangles.

Tangle Tabulation

In addition to classifying tangle diagrams, my research is concerned with the tabulation of tangles. One way to enumerate an exhaustive list of tangles is by generating all possible diagram notations, a method employed by Dowker and Thistlewaite to tabulate knots [14]. This technique for tabulating tangles based on a list of Dowker codes has been used in previous student research projects [21, 11] that I have now taken over and expanded upon in my graduate studies. However, the same tangle may be realized by multiple diagrams and hence this method is known to produce many redundancies. While the resulting list of tangles will be exhaustive, identifying and eliminating redundant tangle diagrams is a challenge for this method of tabulation.

My classification algorithm provides a partial solution to this challenge. Unlike rational and Montesinos tangles, generic tangles are not known to have a preferred diagram. However, every tangle diagram can be decomposed into rational subtangles. Therefore, we can eliminate many redundancies by restricting to diagrams where all rational subtangles satisfy this preferred form. By converting the list of generated Dowker codes into subtangle planar diagram notation, the classification algorithm can identify many redundant diagrams by computing this decomposition into rational subtangles.

I have successfully used this technique to catalog small tangles. For example, every 2-string tangle with fewer than 6 crossings is either integer, rational, or Montesinos, and hence admits a preferred diagram. Figure 6 shows a table of these diagrams for all 5-crossing tangles, excluding mirror images. As crossing number increases, the number of generic algebraic and non-algebraic diagrams grows exponentially, but this technique can still be used to eliminate many redundancies. For tangles with fewer than 10 crossings, the number of remaining diagrams is small enough that we may be able to leverage other methods such as computing tangle invariants to determine potential redundancies for further investigation.

Verification of these results is also an important aspect of this research. The tangles generated using this method are consistent with existing lists of tangles such as [20, 25, 18]. A second method of verifying this output is a bottom-up approach based on generating all preferred diagrams for special families of tangles. By enumerating an exhaustive list of diagram notations for rational, Montesinos, and generalized Montesinos tangles, I have confirmed that my tabulation method does not miss any tangles in these special families. In the future, I will extend this comparison technique to non-algebraic diagrams using constellations.

Database Development

To make my results accessible to the research community at large, I am now developing an online database of 2-string tangles using SQL, PHP, and HTML. By merging my classification and tabulation algorithms, I have written a program in C/C++ that outputs an exhaustive list of notations for tangle diagrams up to a fixed crossing number, including the subtangle decomposition. As a proof of concept, I have created a working prototype of my planned database to demonstrate the functionality of this resource for the special families of algebraic tangles [6]. The next step is to expand this prototype into a full database with the output from the tabulation project.

Some planned features currently under development include: populating the database with tangles up through 9 crossings at minimum; incorporating computer generated figures using KnotPlot [27]; creating a details page for each tangle including notations, construction information, and known invariants; creating an intuitive search form for generating sorted lists of tangles. I am developing this database concurrently with writing my dissertation and will continue to implement features as I make progress.

Future Work

As a postdoctoral researcher, I am interested in translating my skills to new projects. In addition, I have a number of ideas for expanding on my work with tangles. The database I have been developing can be applied more broadly to computational questions in knot theory. Using closures, I can convert the output from my database into a list of knots and links, which can be used to calculate distance tables between knots and to solve tangle equations. Furthermore, subtangle decompositions can also be applied to knot and link diagrams, and I would like to leverage such a decomposition to compute invariants. More generally, I would like to explore how this database can be applied to existing questions in knot theory.

My second idea for a new research direction is motivated by the many applications of tangles to mathematical biology. It is often challenging for scientists to identify the structure of knotted DNA given the difficulty in obtaining detailed images. Image classification is a natural question for computer vision, a topic which I have become interested in through recent internship experiences. Distinguishing between pattern and noise often requires clever data engineering. Given my expertise with tangles and my computational experience in machine learning, this connection to mathematical biology is an excellent opportunity for interdisciplinary collaborations.

Internship Research Experiences

NSF Research Internship with USACE Geospatial Research Laboratory

During the summer of 2020, I participated in a 10-week research internship with the U.S. Army Corps of Engineers' Geospatial Research Laboratory (GRL) through the NSF Mathematical Sciences Graduate Internship Program. The objective of this project was to develop a method for clustering time series data with multi-modal attributes that could take into account similarity with respect to different types of measurements, such as duration, time of day, GPS location, or speed. Working under the direction of Charlotte Ellison, a senior researcher at GRL, I pursued a graph theoretic approach to this question based on a multi-weighted graph model. By leveraging this model to represent multi-modal data, I developed a new graph-based algorithm in Python for clustering data based on a variety of different attributes. This research has been submitted for publication at the 2021 IEEE International Conference on Data Engineering under the title "Multi-modal community detection using multi-weighted graphs" [8].

By quantifying the interaction between pairs of objects, weighted graphs are a flexible tool for identifying patterns such as clusters in data. However, for multi-modal data in particular, different attributes may define different weighted graphs. While each of these graphs models a different attribute, they are defined on the same set of vertices. Thus, we may combine these into a single multi-weighted graph by representing edge weights as vectors. Leveraging this model requires adapting graph-analysis techniques such as community detection (identifying clusters of vertices) to multi-weighted graphs. We developed two novel clustering methods, multi-clustering and meta-clustering, which search for similarity between vertices using a balance of all attributes. Multi-clustering looks at non-empty intersections of clusters from different attribute graphs and is effective when there are few attributes. Meta-clustering is a more robust approach that considers how clusters across all attribute graphs overlap.

Machine Learning Internship with Japanese AI-Startup UsideU

During the summer of 2019, I participated in a 4-week internship with a Japanese company through ICC Consultant's Internship in Japan program. I interned with Tokyo-based startup UsideU, a tech company with a focus on artificial intelligence, on a project in exploratory data science. Working closely with the Chief AI Officer Dr. Alireza Goudarzi, the focus of my project was using deep learning to automate the process of pattern analysis in this company's development of a fitness application. Part of this application involved recording a video of a user exercising and then using computer vision techniques to assess the quality of the performance. The company's existing method for pose-estimation relied on a rule-based system created by fitness experts to determine the position of a user's body, a process we were able to streamline using machine learning.

The result of my project was the creation of a pipeline using Python to automate the most difficult steps in the exercise evaluation process. From a recorded video of a user working through a fitness program, we leveraged computer vision to predict the spatial and angular positions of the most important joins in the body. We then created time series data from each video frame of the recorded exercise which uniquely characterizes a user's performance.

Clustering methods were used to identify distinct positions with an exercise, while PCA and linear regression were used to further analyze patterns in the data representative of a given activity. The overall quality of an exercise is determined by a neural network trained with the data from professional fitness trainers. After the conclusion of my project, I presented my research in a poster session at the Asian Conference on Machine Learning 2019 in Nagoya, Japan [7].

Data Science Internship with American Utilities Company Ameren

Also during the summer of 2019, I participated in a 6-week internship with the utilities company Ameren through the University of Illinois Urbana-Champaign's PI4-IMA Summer Internship Program. I worked as a member of Ameren's interdisciplinary Data Science Team under the direction of senior researcher Dr. Gui Maia on a project applying computer vision to parse important information from PDF documents. The purpose of this project was to automate the process of recording a large backlog of scanned work-orders into an electronic database, a task that was currently being done manually. Using Python, I developed an algorithm to search for and extract key strings of text. This algorithm was able to automate a portion of Ameren's existing record keeping process.

References

[1] Colin C. Adams. *The knot book*. American Mathematical Society, Providence, RI, 2004. An elementary introduction to the mathematical theory of knots, Revised reprint of the 1994 original.

[2] Dror Bar-Natan. Polynomial time knot polynomials, March 2016. Presentation at Advances in Quantum and Low-Dimensional Topology 2016, The University of Iowa in Iowa City, March 11-13, 2016. `http://www.math.toronto.edu/~drorbn/Talks/Iowa-1603/index.html`.

[3] Dorothy Buck and Cynthia Verjovsky Marcotte. Classification of tangle solutions for integrases, a protein family that changes DNA topology. *J. Knot Theory Ramifications*, 16(8):969–995, 2007.

[4] Alain Caudron. *Classification des nœuds et des enlacements*, volume 4 of *Publications Mathématiques d'Orsay 82 [Mathematical Publications of Orsay 82]*. Université de Paris-Sud, Département de Mathématique, Orsay, 1982.

[5] Nicholas Connolly. Graduate research github repository, 2019. `https://github.com/Nicholas-Connolly/Graduate-Research`.

[6] Nicholas Connolly. The tanglenomicon: 2-string tangles database, 2019. `http://www.nick-connolly.com/tangles/tangles.php`.

[7] Nicholas Connolly. Using artificial intelligence to automate body movement analysis, November 2019. Poster Presentation at the Asian Conference on Machine Learning 2019 workshop on Statistics & Machine Learning Researchers in Japan;

Nagoya, Japan; November 2019. https://sites.google.com/view/statsmljapan19/accepted-posters.

[8] Nicholas Connolly and Charlotte Ellison. Multimodal community detection using multi-weighted graphs. In *2021 IEEE International Conference on Data Engineering*, 2021. Submitted.

[9] J. H. Conway. An enumeration of knots and links, and some of their algebraic properties. In *Computational Problems in Abstract Algebra (Proc. Conf., Oxford, 1967)*, pages 329–358. Pergamon, Oxford, 1970.

[10] Peter R. Cromwell. *Knots and links*. Cambridge University Press, Cambridge, 2004.

[11] Isabel K Darcy, Jeff Chang, Nathan Druivenga, Colin McKinney, Ram K Medikonduri, Stacy Mills, Junalyn Navarra-Madsen, Arun Ponnusamy, Jesse Sweet, and Travis Thompson. Coloring the mu transpososome. *BMC bioinformatics*, 7(1):435, 2006.

[12] Isabel K. Darcy, John Luecke, and Mariel Vazquez. Tangle analysis of difference topology experiments: applications to a Mu protein-DNA complex. *Algebr. Geom. Topol.*, 9(4):2247–2309, 2009.

[13] Helmut Doll and Jim Hoste. A tabulation of oriented links. *Math. Comp.*, 57(196):747–761, 1991. With microfiche supplement.

[14] C. H. Dowker and Morwen B. Thistlethwaite. Classification of knot projections. *Topology Appl.*, 16(1):19–31, 1983.

[15] C. Ernst and D. W. Sumners. A calculus for rational tangles: applications to DNA recombination. *Math. Proc. Cambridge Philos. Soc.*, 108(3):489–515, 1990.

[16] Bruce Ewing and Kenneth C. Millett. Computational algorithms and the complexity of link polynomials. In *Progress in knot theory and related topics*, volume 56 of *Travaux en Cours*, pages 51–68. Hermann, Paris, 1997.

[17] Jim Hoste, Morwen Thistlethwaite, and Jeff Weeks. The first 1,701,936 knots. *Math. Intelligencer*, 20(4):33–48, 1998.

[18] Taizo Kanenobu, Hirofusa Saito, and Shin Satoh. Tangles with up to seven crossings. In *Proceedings of the Winter Workshop of Topology/Workshop of Topology and Computer (Sendai, 2002/Nara, 2001)*, volume 9, pages 127–140, 2003.

[19] Louis H. Kauffman and Sofia Lambropoulou. On the classification of rational knots. *Enseign. Math. (2)*, 49(3-4):357–410, 2003.

[20] Louis H. Kauffman and Sofia Lambropoulou. On the classification of rational tangles. *Adv. in Appl. Math.*, 33(2):199–237, 2004.

[21] Ram Kishore Medikonduri. *Tabulation of tangles and solving tangle equations*. ProQuest LLC, Ann Arbor, MI, 2007. Thesis (Ph.D.)–The University of Iowa.

[22] William Menasco and Morwen Thistlethwaite. The classification of alternating links. *Ann. of Math. (2)*, 138(1):113–171, 1993.

[23] Markus Meringer. Fast generation of regular graphs and construction of cages. *Journal of Graph Theory*, 30(2):137–146, 1999. `http://www.mathe2.uni-bayreuth.de/markus/reggraphs.html#PLANAR`.

[24] Hyeyoung Moon. *Calculating knot distances and solving tangle equations involving montesinos links*. ProQuest LLC, Ann Arbor, MI, 2010. Thesis (Ph.D.)–The University of Iowa.

[25] Hyeyoung Moon and Isabel K Darcy. Tangle equations involving montesinos links.

[26] Dale Rolfsen. *Knots and links*, volume 7 of *Mathematics Lecture Series*. Publish or Perish, Inc., Houston, TX, 1990. Corrected reprint of the 1976 original.

[27] Robert Glenn Scharein. *Interactive topological drawing*. ProQuest LLC, Ann Arbor, MI, 1998. Thesis (Ph.D.)–The University of British Columbia (Canada).

[28] Ying-Qing Wu. The classification of nonsimple algebraic tangles. *Math. Ann.*, 304(3):457–480, 1996.

[29] H Yamano. Classification of tangles of 7 crossings or less. *Master Thesis, Tokyo Metrop. Univ*, 2001.

[30] Heiner Zieschang. Classification of Montesinos knots. In *Topology (Leningrad, 1982)*, volume 1060 of *Lecture Notes in Math.*, pages 378–389. Springer, Berlin, 1984.