Interactive Topological Drawing

by

Robert Glenn Scharein

B.Sc. (Physics, Honours), The University of Manitoba, 1981
M.Sc. (Astronomy), The University of British Columbia, 1985

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

Doctor of Philosophy

in

THE FACULTY OF GRADUATE STUDIES

(Department of Computer Science)

The University of British Columbia

March 1998

Abstract

The research presented here examines *topological drawing,* a new mode of constructing and interacting with mathematical objects in three-dimensional space. In topological drawing, issues such as adjacency and connectedness, which are topological in nature, take precedence over purely geometric issues. Because the domain of application is mathematics, topological drawing is also concerned with the *correct* representation and display of these objects on a computer. By correctness we mean that the essential topological features of objects are maintained during interaction.

We have chosen to limit the scope of topological drawing to knot theory, a domain that consists essentially of one class of object (embedded circles in three-dimensional space) yet is rich enough to contain a wide variety of difficult problems of research interest. In knot theory, two embedded circles (knots) are considered *equivalent* if one may be smoothly deformed into the other without any cuts or self-intersections. This notion of equivalence may be thought of as the heart of knot theory.

We present methods for the computer construction and interactive manipulation of a wide variety of knots. Many of these constructions would be difficult using standard computer-aided drawing methods. Interactive techniques allow for knot simplification under topological constraints from complicated conformations to simpler embeddings. These methods have proven useful in the investigation of the knot equivalence problem.

As a further test of its utility, topological drawing has been used for several knot theoretical applications. The first of these involves finding the *stick-number* of a knot (the fewest number of straight sticks needed to form the knot). A second application is to the relaxation of knots

under a physically-based knot energy (the symmetric energy) that we find effectively simplifies knots to configurations approaching their "canonical form". Finally, our methods have proven useful in the visualization of a class of knots that arise in a study of three-manifold topology. These knots often have complex descriptions (for example, as a huge braid word), but may be simplified greatly through the use of interactive topological drawing. Here, an expert user relies on the visualization in order to steer the computation in a direction that will often significantly improve performance.

Contents

List of Tables

List of Figures

List of Colour Plates

Page numbers indicate the facing page.

Acknowledgements

> remember patience is the great thing. . .
>
> James Joyce, *Finnegans Wake*

Requiring more than sixteen years to complete graduate school demands an explanation and an apology. I shall provide neither. This space is for acknowledgements, so here we go. . .

I am exceedingly grateful to my supervisors, Dr. James Little and Dr. Kellogg Booth for their support over the years. Especially near "the end", they were crucial in helping me stay afloat long enough to produce a thesis. To my entire committee, Jim, Kelly, Dr. Dale Rolfsen and Dr. Nicholas Pippenger, I am thankful for much inspiration and guidance.

Many friends have been important over these years, those in the department: Scott, Bill, Dan, Valerie, Gwen, Carl, and Chris, those in Vancouver: Nedenia, Gavin, Cameron and Adele, and those farther away: Rob, Eric, Tom, André, Ivan, Ray and Dave (hope I didn't forget anyone). You are all treasures in one way or another, I have been gifted to know you all. Added to these friends are a number of gracious felines that have crossed my path over the years: Iris, Maggie, Pepper, Hornby, Loki, and Kopek.

For financial support, I wish to thank the Natural Science and Engineering Research Council of Canada, the British Columbia Advanced Systems Institute, the Institute for Robotics and Intelligent Systems Network of Centres of Excellence, the TeleLearning Network of Centres of Excellence, the Province of British Columbia through support for the Media and Graphics Interdisciplinary Centre, and The University of British Columbia.

My family has been very supportive and loving. I thank my brother Don, my sister-in-law Debbie, and my wonderful nieces, Jessica, Stephanie, and Rachel for countless moments of great joy in their company. It is to them that this thesis is dedicated.

ROBERT GLENN SCHAREIN

The University of British Columbia

March 1998

To Don, Debbie, Jessica, Stephanie, and Rachel

Chapter 1

Introduction

A fundamental issue in mathematics is determining when two objects are equivalent. This problem may be approached from many different angles and refined to an almost unlimited degree. One field of mathematics where questions of this sort abound is topology. Topological notions of equivalence are based on fundamental, proto-geometric ideas such as adjacency and connectedness. Two objects may have the same "shape" in a topological sense even if they appear at first glance to be quite different. For example, consider the objects in Figure 1.1, where the two curves represent loops of string in three dimensional space. Are these shapes
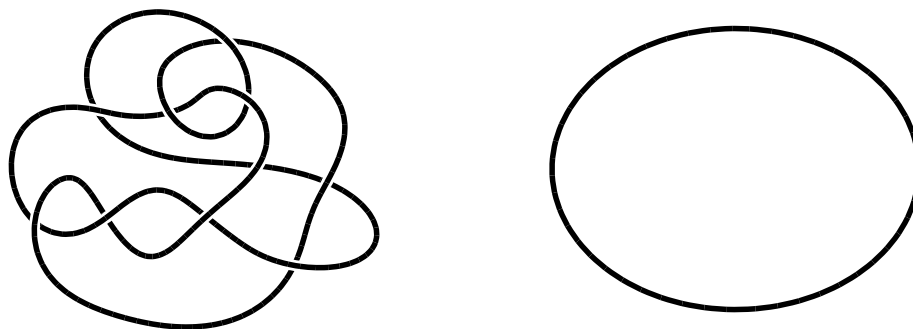
Figure 1.1: Two objects with the same shape.

equivalent? Here, equivalence is meant in a broad sense: two such shapes are equivalent if and only if one may be continuously deformed into the other without any cuts, self-intersections, or

singular points in the objects. The material is assumed to be ideal and infinitely deformable. Surprisingly, even this apparently simple case of two loops in space has proven to be an extremely difficult problem. The sub-branch of topology concerned with problems of this sort, knot theory, has developed over the past century a powerful array of techniques that have gone a long way in addressing such equivalence problems. Although these matters have been resolved in principle, *i.e.* the problem is known to be decidable, the solutions suffer from being highly impractical (in a computational sense) even for simple examples.

Let us suppose that we know the above two shapes are equivalent in the sense described (they are). How does one go about demonstrating this equivalence, say by exhibiting a deformation that takes one shape into the other? In knot theory, this is a computationally difficult problem. This fact may appear somewhat surprising. In contrast, in the field of computer graphics, impressive claims are occasionally made for what is known as the "morphing" problem (exhibiting a smooth transition or metamorphosis between two different objects). Although most researchers in the area are well aware of the difficulties of morphing between two arbitrary three-dimensional objects [Hug92], DeCarlo and Gallier [DG96] point out that many do not take topology seriously. The fact that all the power of modern mathematics is yet unable to effectively "handle" cases such as Figure 1.1 indicates that if we do take topology seriously, we are essentially stuck at the level of one of the simplest imaginable problems in 3D morphing.

In fact the problem is even more difficult than we have so far indicated because it contains an additional subtlety. It is possible to create a smooth transition between the objects in Figure 1.1 as we shall see in Chapter 7. However, if a small *local* change is made to the initial object to arrive at the new object shown on the right in Figure 1.2 then the smooth transition to a circle is *no longer possible.* This demonstrates that it is necessary to take the *global* topology of an object into account, not only to generate a transition into another object, but even to determine whether such a transition is possible at all.
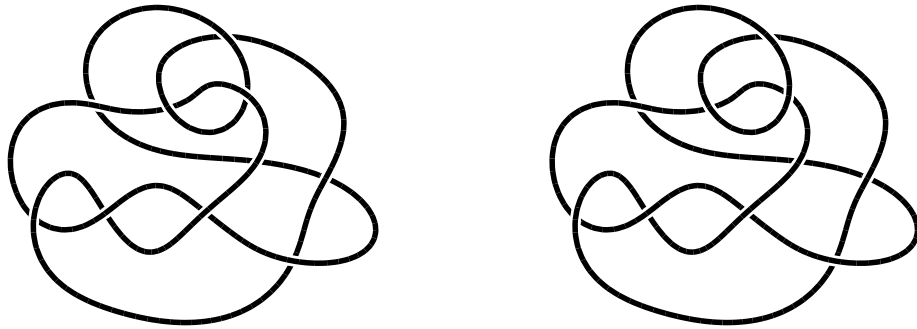
Figure 1.2: Two objects with a different shape.

1.1 Topological drawing

The discussion above suggests that a new approach must be taken where issues of a *topological* nature take precedence over purely geometric ones. *Topological drawing* focuses on notions such as adjacency, connectedness, and the topological character of the embedding or placement of an object in space; it is relatively unconcerned with precise geometrical placement. Dennis Roseman [Ros92a, Ros92b] was probably the first person to use the term topological drawing with reference to a *mathematician's drawing tool.* The requirements of such a tool are somewhat more stringent than ordinary drawing systems, where issues of correctness are not generally a great concern.[1] Roseman's design goals are fairly broad [Ros92a] and target mathematical constructs considerably more complicated than anything of concern to this thesis. We will limit the scope of our study by selecting only a subset of topological drawing.

The arena chosen for demonstrating topological drawing is knot theory, for several reasons. First of all, knot theory can be considered to be the simplest, nontrivial example of a *placement problem* [Fox62]. Other placement problems (motion control in robotics, human body movement) inherit some of the difficulties already found in knot theory. Secondly, although knot theory is a beautiful branch of modern mathematics and can certainly stand on its own merits, connections are increasingly being found with other sciences, such as physics, chemistry and biol-

---

[1]This is true to a limited extent, of course. In applications that rely on computer aided design and manufacturing, correctness can often be a life and death matter. However, from a mathematical point of view these systems cannot be regarded as "rigourous".

3

ogy. Some of the applications of topological drawing explored in this thesis are directly targeted in these directions. A final reason for studying knots is the intuitive nature of knot theory. Knots are something most of us work with everyday and they have practical, non-mathematical applications [Ash44, GH52, Lee74, vdG92] as well as a rich cultural history [Con92, vdG93, TvdG96]. It is intriguing that many problems in knot theory are easily phrased, yet they have difficult solutions or may even be unsolvable in practice. Another aspect of knots is their long and distinguished presence in the decorative arts [Bai73, Bac81, Bai86, Mee91], dating back well over a millennium. One look at the Lindisfarne Gospels, a medieval illuminated manuscript from Northumbria [Bac81], is a humbling experience for any would-be drawer of knots because it indicates that there is almost unlimited room for improvement over any existing computer-aided system.

1.2   Mathematical visualization

In addition to providing a foundation for topological drawing, Roseman [Ros95] also outlines a number of principles that apply to any mathematical visualization system. One important principle is the difference between *display* and *visualization.* Essentially, the difference lies between simply creating a picture (display) and allowing the viewer to form a mental construct (visualization) of the object being studied. Roseman provides an apt quote from Gibson [Gib79]:

> "A surface can be seen; a plane can only be visualized."

A mathematical plane is an abstract concept that cannot be "seen" but must be imagined by combining a visual impression together with an intellectual understanding of the mathematics (in this case the mathematics of two-manifolds). A display can be considered a visualization if it aids in this understanding.

In this thesis we will not be overly concerned with the philosophical issues of mathematical visualization. It is hoped that the reader will agree that the images appearing here are indeed visualizations and not merely "messy displays". For greater insight on the matter of mathematical visualization, the reader would be advised to seek out the paper by Roseman [Ros95]

and the references it contains. Other excellent sources are the books by Hilbert and Cohn-Vossen [HCV52] or Reichenbach [Rei57]. The collection of articles edited by Brisson [Bri78b] are "early" works in the field of mathematical visualization by computer.

1.3   Contributions of the thesis

"Your work is interesting; it's not computer graphics, but it's interesting."
— *Critique of the thesis work by a famous computer graphics researcher*

The thesis research, combining both topological drawing and mathematical visualization, has resulted in the creation of useful and powerful techniques for the exploration of numerous problems in mathematical knot theory and related fields, such as the physics of knots, DNA biology, and topological chemistry. This thesis is novel in the sense that it is concerned with both the development of the computer aided visualization techniques themselves, a domain within the bounds of computer science, as well as the application of those techniques to areas outside of traditional computer science. The vehicle for this exploration has been the computer program KnotPlot, written by the author.

The remainder of this section highlights the scientific contributions of the thesis work to the areas of study in computer science (primarily computer graphics) and mathematics. In addition to these "main-stream" disciplines, the thesis research has also found artistic applications, some of them quite far removed from computer science or mathematics.

1.3.1   Computer graphics

Despite the above quoted comment, this thesis does contribute to computer graphics in several important ways. The first of these contributions concerns the issue of *correctness*. Computer graphics is a field that, for good or for ill (depending on one's point of view), has been characterized by the notion that "if it looks good then it must be right". This thesis maintains that for certain applications it is important to go beyond this. That is, it is necessary to produce pictures that are *provably correct* renditions of what they purport to be. An analogy may be drawn

here with symbolic algebra systems such as Maple [Mon96] and Mathematica [Wol91], which are increasingly used in mathematics research. Experience in their use is now a standard part of a mathematician's training at many universities. Issues of correctness for symbolic mathematics are probably somewhat more severe than we will find necessary in topological drawing, largely because of the vast number of (real world) applications that depend on this correctness.

In a similar vein, there has been some neglect of correctness issues when modelling interactions between computer generated objects. For example, interpenetrating or self-intersecting objects are common occurrences in computer animations. Generally this is not a problem because the deviations from correctness usually cannot be seen during the animation, and they in no way affect the visual quality of the finished product. In contrast, this thesis is concerned with objects that have an existence as pure mathematical entities; if the results of manipulation or interaction are to be believed, it is essential that they be provably correct. This approach is similar to interactive, physically-based modelling systems for proteins [Sur92b, Sur92a], where correctness to the underlying biochemical reality is imperative.

On the other hand, it is important not to overstate these issues. Even if a claim of correctness for a software system is made, mathematicians will *still* check the results using other methods, assuming such a method is available. The reason for this is simple: a software implementation is exceedingly difficult to verify absolutely (*i.e.* one that solves non-trivial problems of actual complexity). Also, the internal structure of complicated software is usually not amenable to inspection to the same degree as a mathematical proof on paper. There is always the risk of error. Software designers can do their best to reduce the risk of error to a sufficiently low level so that users will have a degree of faith in the system. If the system is too complicated to verify, or if a mathematical proof relies too heavily on computerized methods, a substantial fraction of mathematicians may find it takes less faith to simply believe the theorem outright than it does to believe the proof.[2]

Finally, we make the case that topological drawing is a powerful and novel method for computer assisted modelling. Although apparently similar work has appeared before [FRC92],

---

[2]A case in point being Appel & Haken's proof of the four colour theorem [AH89].

the applications were rather simple and resulted in models that could have been easily designed with more traditional means. More sophisticated surface modelling tools, such as Welch's "topological putty" [WW92, Wel94] and other variational modelling techniques [MS92], are not topological drawing in the sense meant in this thesis, as their intent is to be *geometric* modelling tools. That is, the user has in mind a specific geometric form and wishes to use the system to manipulate the surface to satisfy imposed geometric constraints. With topological drawing in our sense, the user gives up all (or almost all) control over the geometric placement. Rather than having a specific design goal as in variational modelling, in topological drawing the resulting surface is a *side effect* of a simpler underlying mathematical object, and is *essentially arbitrary.* That it may prove useful in creating complex and unusual surface models (see the examples in Chapter 8) is a pleasant outcome, but not the primary objective.

1.3.2 Mathematics

While the mathematical content of this thesis is elementary compared to the state of the art in mathematics research, mathematicians (knot theorists) can benefit from the interactive techniques presented here. Part of this is empowering them to be able to construct, manipulate and transform their objects of study easily. Methods for topological drawing can be used by mathematicians as an aid in their research. The computer system acts as a bookkeeping and consistency checking device. Some operations and calculations formerly done by hand may now be performed easily by computer. This is similar to other work in topology that uses computers to simplify the complexity of calculations [Rav89]. It is important to emphasize that intelligent intervention by persons with some understanding of the subject area is often crucial to the success of such systems.

An important theme that recurs throughout this thesis is the quest for a *canonical form* or *optimal shape* for knots. This topic is of considerable theoretical interest [O'H91, Sim94, FH92, KS94, BSS98a] at the present time, with close ties to both physics [Mof92a, Sch92, LS94, BSS98a] and biology [Sum85, KBM+96, KOP+97]. In these approaches, the canonical form is considered to be one that minimizes some energy functional of the knot, assuming that such a minimum

exists (and that it is unique). Depending on the energy functional chosen, different "canonical forms" may result. There is not universal agreement on what constitutes a canonical form for knots (with one or two exceptions). Each of the energy functionals studied by researchers has its merits. In this thesis, we will examine only a small number of these energy functionals, however, the techniques developed here make it possible for other researchers to investigate new methods or formalisms.

In addition to finding canonical form, research presented in this thesis has focussed on other open problems in knot theory, such as computing the stick number of a knot [Ran94, Mei96, ABGW97] or the problem of knot recognition (the equivalence problem that opened this chapter). Our results compare well to similar work in these areas [Mei96, KS94, LS94, GHK97]. For finding stick numbers, our method is somewhat of a "brute-force" approach in comparison to the work of Meissen [Mei96], however, the work load is transferred to the computer rather than the human. The second of these areas, that of knot recognition, might be considered a subset of the canonical form problem; *i.e.* if the canonical form of a knot can be found, then all we need to do is to compare the canonical forms of two knots and see if they're the same. We have grouped it here as a separate problem because some of our methods step outside the energy-based approach. They take advantage of the fact that we are "doing topology" and not physics; therefore operations can be performed that would be "illegal" in a physically-based context. This may make these methods seem *ad hoc* in comparison to the sound theoretical basis of, for example, the work of Kusner and Sullivan [KS94]. We defend our position by saying that if the problem of knot recognition is to be computationally addressed, then all mathematically possible techniques should be used, and not just those that have some justification in a specific physical model.

The mathematics literature contains numerous examples (of which [Fen89, FR86] is a sample) of classes of mathematical objects that have never been interactively displayed on a graphics workstation. It has been shown that useful information can be extracted from such exercises in certain cases [KBBL86, MMHB93, JS94], or at least such visualizations can contribute to the understanding of the mathematics involved. To this end, the thesis research has been

useful in the visualization of a non-trivial theorem in topology [Sed96] as well as in the visualization of knots related to a census of hyperbolic three-manifolds [Dea97]. The latter case was a particularly good test of our methods. Interactive manipulation provided guided relaxation of complicated knots to simpler forms under topological constraints.

Finally, the thesis work also provides powerful tools for many practical applications in knot theory; *i.e.* the "day-to-day" tasks. Mathematicians are able to create complex examples of many types of knots easily and interactively. Extensive support is provided for the production of high quality illustrations of these examples. These illustrations have appeared in several publications [Sch91, Wu92, Sch95, KBM$^+$96, MSS97, Dea97, Sch97a, KG97, KOP$^+$97, CDW98].

1.3.3 Art and design

Work with the mathematical knot visualization program KnotPlot (see Chapter 4 and [Sch97b] for detailed information) has attracted the interest of both artists and non-artists for purely aesthetic reasons [Bot94, Gib94, HBK$^+$94, New96]. While such applications may appear at first glance to be somewhat outside the scope of a thesis in computer science, they are a central driving force behind the work described herein. Many mathematicians (as well as other scientists) are motivated by an *aesthetic* appreciation for their work: "Mathematics is the finest of the fine arts" [Lan85]. The methods of topological drawing with automatic refinement have proven useful in generative design. These applications are in the same spirit as contemporary developments in "mathematical art", which has only recently incorporated more sophisticated mathematical content [FC92, Cox92, Nol94].

1.4 Overview of the thesis

A brief history and review of mathematical visualization is given in Chapter 2. Chapter 3 contains the background in knot theory relevant to understanding the problems addressed in the thesis. It is of necessity a cursory overview, as the field is vast and ever expanding. Included in Chapter 3 are the definitions of concepts in knot theory that will be discussed in later sections of

the thesis as well as a survey of research done by others in the field of knot relaxation, a key area addressed by this thesis. Chapter 4 introduces KnotPlot, our vehicle for exploration of topological drawing. Most of KnotPlot's features are deferred to later chapters; Chapter 4 provides a general overview of the system and various issues not directly related to topological drawing, such as graphical display, topological and geometrical calculations, and the construction and display of higher-dimensional objects.

Chapters 5 and 6 are concerned with providing initial embeddings of knots and links using a number of different techniques. Chapter 5 presents a large array of methods for the creation of knots and links. These methods are designed to interface readily with each other; often several simple techniques combined can yield a non-trivial result. The technique described in Chapter 6 is somewhat different. It addresses one specific problem, that of constructing an embedding of a knot given an encoding (in the form of a list of integers) of a diagram of the knot. The two chapters are complementary in some ways, Chapter 6 providing an alternative to the main approaches used in the thesis. The methods discussed in Chapters 5 and 6 often create embeddings that are rough and "unpolished". How these are improved to an acceptable final form is the subject of Chapter 7, which discusses topological refinement, both automatic and interactive.

Chapter 8 examines applications of topological drawing to research problems in knot theory. A summary and suggestions for future work concludes the thesis in Chapter 9.

There are a few stylistic conventions used in this thesis that should be noted at this point. Citations of literature earlier than 1900 are indicated by placing a bar over the year number. Thus, Tait's 1876 paper "On Knots" is cited as [Tai$\overline{76}$]. Two simple theorems appear in the thesis, the ends of their proofs are indicated by the symbol ⬡.

Chapter 2

Mathematical visualization

This chapter is a general review of previous work in mathematical visualization. To be fair, most of modern mathematics can be considered mathematical visualization. Mathematicians are experts at communicating subtle ideas of great complexity. Here we will focus on work that is primarily graphical in nature, especially if it involves visualization with the aid of a computer. Also in this chapter is a brief overview of interaction in 3D and 4D and the specific problems associated with the visualization of objects in higher dimensions.

2.1   Mathematical visualization in the traditional media

As we have stated, if we consider diagrams acting as visual aids, then almost any mathematical paper might be considered a form of mathematical visualization. To find examples we might go back as far as Gauss, Euler, or even Euclid. However, generally these diagrams are not intrinsic to the *proof* of theorems; *i.e.* the diagrams could be deleted without any effect on the correctness of the results. It seems more appropriate therefore to limit the scope of mathematical visualization to be those areas where the interesting features of the study are *primarily visual*. In this stronger sense, mathematical visualization started in this century when mathematicians began to take special efforts to convey their ideas in a highly visual manner. One of the early works using this approach is the 1932 German classic *Anschauliche Geometrie* by David Hilbert and S. Cohn-

Vossen, available in English translation as *Geometry and the Imagination* [HCV52]. This book is remarkable for the quality of insight and also for its incredible diagrams, mostly hand drawn and (obviously) predating the computer era. These diagrams are useful to current mathematical illustrators as examples of good form in depicting a variety of mathematical objects.

The Hypergraphics symposium was held in 1978 and brought together artists, architects, mathematicians, and computer programmers in order to share insights on higher-dimensional geometry and its graphical representation. The proceedings, published as a book [Bri78a], contain a number of interesting reflections on the visualization of higher dimensional objects [Bri78b, Nol78, BS78] such as hyper-cubes and also an impossible 4D illusion constructed by Scott Kim [Kim78].

Several books have been published in the past decade that are superb examples of mathematical visualization. Perhaps the best of these is *A Topological Picturebook*, George Francis' beautiful book [Fra87] on many different aspects of topology (other work by Francis is equally impressive, for example [FC92]). Most of the illustrations are hand-drawn diagrams of 3D objects, but he does include a chapter on higher-dimensional entities such as the projective plane. Francis is an expert at drawing *Seifert surfaces*. These are orientable surfaces that span a knot (*i.e.* they have the knot as a boundary [Rol76, Ada94, Mur96]). Equally beautiful, but intended for a more general audience is Thomas Banchoff's recent book *Beyond the Third Dimension* [Ban90, BC92], which popularizes some mathematical concepts and contains numerous high quality illustrations of 4D objects.

As was discussed in Chapter 1, a visualization goes beyond mere display if it serves to elucidate the concept in mathematics that it is depicting. The paper by Hart, Francis, and Kauffman [HFK94] provides an effective visualization of quaternion algebra and its relationship to rotations in 3D. Particularly interesting is the illustration of the counter-intuitive "Dirac string trick" (also known as the "belt trick" [MTW73, Han89, Kau91]). One feature of the quaternion demonstrator is that a person can "try it at home" (see [Kau91] to find out how to make one).

2.2 Computer systems for mathematical visualization

This section discusses software specifically designed to be general purpose mathematical visualization systems. Much of this work has come out of the Geometry Center (funded by the U.S. National Science Foundation), associated with the University of Minnesota. Fortunately for researchers much of this software is free, and source code is often available.

An early attempt at producing a comprehensive mathematical visualization package was undertaken by David Dobkin's group at Princeton. They produced Salem [DNT90, DNT91], a viewer for mathematical objects in 3D. These objects are all constructed of sets of points (vertices) and polygons. From these primitives, Salem could construct higher-level objects, such as polyhedra (*i.e.* the named polyhedra with regular faces [Joh66, Zal69]), tori with arbitrary numbers of holes, convex hulls of point sets, and torus knots. Salem came packaged with a well-designed interface to the underlying UNIX operating system, providing both direct access to the file system, as well as an ability to create user-defined clients, which could then be used to generate arbitrary objects. Salem provided a full set of commands for object and view transformation, some of it under mouse control. Perhaps the most interesting aspect of Salem was its ability to navigate through elliptic or hyperbolic space and to illuminate surfaces there.

Perhaps the premier mathematical visualization software package anywhere is the Geometry Center's GeomView [PLM93]. GeomView, like Salem, can display objects in spherical or hyperbolic space. GeomView has most of the other features of Salem (in some ways GeomView might be considered to be a successor to Salem). GeomView is also freely available from the Geometry Center. As a result it is widely used by many geometers, physicists, and other people for many visualization tasks.

Finally, a claim could be made for the use of the general purpose systems Maple [Mon96] and Mathematica [Wol91] in mathematical visualization. However, these systems are not really mathematical visualization packages in the same sense as Salem or GeomView, coming as they do with little or no "knowledge" of geometry or topology. Also, neither system allows for interactive manipulation of mathematical objects.

2.3 Film and video

One of the first films attempting to illustrate a non-trivial theorem in mathematics was Nelson Max's 1977 film *Turning a Sphere Inside Out* [Max77]. The theorem was originally due to Smale and proves that an ordinary two-sphere (such as the surface of the Earth) can be turned "inside out" (everted) in a smooth way (see the article by Phillips [Phi66] for a readable account). Obviously, if the smoothness restriction is dropped, it is simple to turn a sphere inside out: simply grab opposite points of the sphere and pull it through itself. At some point this method will form a "crease", for example along the equator of the sphere. Smale's proof shows that there exists a smooth eversion of the sphere; *i.e.* one without creases. In order to avoid creases, the sphere must be subject to amazingly complicated contortions. Unfortunately, the original film by Max is not readily available, but the reader is urged to consult the earlier article by Phillips [Phi66] in *Scientific American* which, although it illustrates a different proof, gives an idea of the intricacies involved. Recently a team at the Geometry Center has produced a new version of a sphere eversion animation, a video called *Outside In* (see the cover of *Scientific American*, October 1993). This animation is highly effective in illustrating the eversion process, making it appear almost "obvious". However, in all fairness to the pioneering effort by Max, this "re-make" is more successful than the original largely due to vast improvements in computer power, better software, and a much larger production team with more years of experience making computer animations. Also, perhaps more importantly, it illustrates a completely different (and much simpler) proof of the sphere eversion (due to Thurston).

Another Geometry Center production is Charlie Gunn and Delle Maxwell's beautiful animated film *Not Knot* [GM91] visualizing the hyperbolic structure associated with the Borromean rings.[1] Gunn has also published related work [Gun93] on the visualization of discrete groups and three-manifolds (see also [PG92]). Visualizing the hyperbolic structure associated with a knot is a much more difficult visualization problem than sphere eversion as the action takes place in an unfamiliar realm (hyperbolic three-space) as opposed to ordinary three-dimensional

---

[1]This is a famous link of three unknotted circles any two of which alone are unlinked.

14

space, although perhaps the structures are fairly simple ones. *Not Knot* used much of the same software (GeomView and various other software packages, some of them custom written for each project) as did *Outside In*.

2.4   3D interaction

Intrinsic to any mathematical visualization system is the ability to navigate and manipulate effectively in whatever space or structure is under study. This section briefly describes some of the work in 3D interaction that pertains to the thesis. One problem in computer graphics is how to specify an operation that is intrinsically three-dimensional using an apparently two-dimensional input device such as a mouse. It may seem paradoxical that there is no limit to the number of dimensions that a mouse can be used to specify. This is because, although the *location of the mouse cursor* is constrained to a two-dimensional surface (the computer screen), the *space of paths* through this two-dimensional surface has *infinite dimension*[2] (imagine an infinite procession of sliders, one for each dimension). Therefore, in principle there is nothing that a mouse could not be used to specify, however, it may require users to perform more and more intricate maneuvers with the mouse.

A simple task that is easily performed through mouse movements is the specification of an orientation in 3D. This is usually done by imagining a *virtual input device* that emulates a real physical device used to directly specify 3D orientations, for example a trackball. These "virtual spheres" attempt to emulate a real physical trackball consisting of a sphere held in such a way that it can rotate freely without translating [CMS88, Sho92, Han92]. Trackballs are often found on video-games but are not a common device on most computer systems. A trackball can be emulated in a natural way because of the lack of *commutativity* inherent in 3D rotations. This means that a sequence of rotations AB is in general not undone by the sequence $A^{-1}B^{-1}$. According to Schutz [Sch90] this lack of commutativity is necessary if infinitesimal operations in a two-dimensional space are to produce a net result that leaves that space.

There has been some discussion in the literature about which formulation of a 3D trackball

---

[2]Ignoring, of course, the fact that there are only a finite number of cursor positions.

is the most intuitive or useful, or the most efficient at performing specific tasks [Sho92]. It seems that this is largely a matter of personal taste and the degree of training and familiarity with each system. All of the trackballs described in this section appear to be within "error limits" of each other for most tasks. They are all clearly better than using three separate linear sliders to specify Euler angles. All of them have some degree of physical realism to them, and all are easily mastered in a matter of moments. In the final analysis it probably doesn't matter mathematically, as there is only one algebra of 3D rotations.

Chen's work [CMS88] is apparently the first implementation of a virtual trackball in the literature. It is no doubt wide-spread by now. Implementing it from the description in the paper can be tricky because of a number of ambiguities and errors in the explanation. Also the authors give no justification nor citation for a rather mysterious and complicated looking matrix they use. This matrix is none other than the *Rodrigues equation* [Alt86, page 75], which gives a rotation matrix in terms of an angle and a rotation axis. The principle upon which the trackball works is simple. From movements of the mouse it is possible to derive a rotation axis and an angle. These are used to update the state of the rotation matrix that is part of the modelling transformation of the object being displayed. Because the axis is *continually changing* as the mouse is moved, it is possible to access the entire orientation space.

Shoemake's Arcball [Sho92] seems to be one of the most popular virtual trackballs and is probably the best overall method for any task involving precise control. Many existing 3D modelling tools would stand to benefit by incorporating the arcball. The arcball is also considerably easier to implement than the trackball. Unfortunately, Shoemake appears to be unclear on the difference between hysteresis and commutativity, so his claims are somewhat misleading. Nevertheless his formulation is firmly based on quaternion algebra.

Hanson's "Rolling Ball" [Han92] is possibly the most theoretically satisfying formulation of these variations on a theme. Hanson examines the rolling ball concept from the point of view of Lie group theory. His method extends naturally into four and higher dimensions.

There are other versions of the virtual trackball floating around in the literature; the above are the some of the most well known.

2.5 Visualization in four dimensions

This section surveys some of the relevant literature concerning the visualization of 4D objects and approaches to interacting with them. Much of the work on the display of 4D objects (more precisely, 2D surfaces in $\mathcal{R}^4$) concerns itself with the correct rendering of such surfaces. Because of the lack of a unique normal to such a surface in $\mathcal{R}^4$, the usual methods for surface shading used in computer graphics no longer apply. Also, because the surface must eventually be projected to a 2D screen (and retina), there is a problem with occlusion (which is already serious in 3D). Finally, there is the issue of the severe distortion that occurs when two out of the four dimensions are lost during the projection from 4D to 2D.

In order to produce an image of an object in $\mathcal{R}^4$ on a 2D computer display, one of two methods (or a combination of both) are commonly used. The first of these involves choosing a suitable 4D "eye-point" in a similar manner to 3D graphics. This point is used to produce a perspective projection to 3D using equations that are exactly analogous to the standard 3D to 2D perspective equations. The perspective projection from 4D then produces a 3D object (2D surface in 3D) which may then be displayed using the usual computer graphical techniques. Alternatively, an orthographic projection might be used (this is of course a limit case of a perspective projection). In their investigation of the topology of constant energy surfaces of dynamical systems in 4D, Koçak *et al.* [KBBL86] make a strong point about the difficulty of graphics in 4D. They show an example of the difference a choice of projection from 4D to 3D can make (orthographic or perspective). They find that in general a perspective projection is more informative, and that important features of the surfaces they study can sometimes be completely obscured by an orthographic projection. Another important point in the paper is that animation is a key factor in the visualization of 4D objects. Animating an object undergoing rigid rotations in 4D allows one to see many different projections of the same object.

Often combined with the projection technique is the method of *ribboning*, pioneered by Banchoff [Ban86]. This is the technique of "painting" transparent stripes on a surface. He used ribboning to produce an effective illustration of the Hopf fibration of the three-sphere. Ribboning is an effective way of getting around the problem of occlusion (just as it is in 3D). Also, if the

17

stripes are painted a colour according to the value of the dimension lost in the projection, then surfaces are much easier to visualize.

The second method of displaying 4D objects uses a "slicing" technique, where a 3D manifold (usually a 3D hyperplane) of some sort is used to intersect the 4D space where the object of interest lies. Resulting from this intersection is some subset of the manifold used for the slicing, which is again amenable to display with standard methods in computer graphics. Time can also be used a fourth dimension directly; some objects are effectively visualized as a sequence of 3D slices.

Recent work by Roseman [Ros95, Ros96] focuses on the visualization of knotted 2-spheres in 4D (see Chapter 3). Roseman uses ribboning, projection, and slicing techniques along with animation to illustrate complex deformations of these knots in 4D. He makes a strong point that careful choice of the slicing and projection directions is important when using both techniques together. In particular, it is essential to ensure these directions are not parallel.

Most researchers cited in this section consider interaction as well as display issues. However, the interaction is generally of a *global* nature. That is, rotations, scale changes, and translations are applied to the scene as a whole. Of these three interactions, the most (perhaps only) difficult one is higher dimensional rotation. Scale changes and translations are not significantly different in 4D than they are in 3D. However, rotations are quite another matter. The space of 4D rotations $SO(4)$ is *six-dimensional* as opposed to four-dimensional. Hoffmann and Zhou [HZ90] present a useful discussion of 4D rotations using Euler angles that helps to clear up some of the difficulty. The authors also point out that animation is of crucial importance for visualization in 4D because of the significant amount of information loss that results from the two-step projection from 4D down to the 2D computer screen. Moreover, 4D rotations can be extremely counter-intuitive. A rigid smoothly rotating object in 4D can appear after the projection to 3D to be an object in 3D undergoing distortion.

Much of the work in 4D visualization in the computer graphics literature concerns itself with rendering issues. Ke and Pandurange [KP90] present work on raytracing a 4D analogue of the Mandelbrot set, using animation and slicing techniques. Andrew Hanson and his co-workers

[HH91, HH92, HC93] take a original approach to the illumination of surfaces in higher dimensions. Surfaces in 4D have no unique normal, so he first "fattens" out the surface by attaching a 3-ball to each point. This is similar to the way a 1D object is rendered in a 3D environment. By attaching small disks to each point it is possible to produce an object that has a unique surface normal. Hanson illuminates the surface (now a 3-manifold) in 4D and then projects it to what he calls "3D film". This results in 3D volume data which he then volume-renders using standard techniques in volume rendering. This method produces interesting pictures, but obviously has all the problems of volume visualization. Most computer graphics platforms are still not at the stage where anything approaching interactive speeds can be expected.

There have been a few attempts at creating general visualization systems for objects in 4D. Beshers and Feiner [BF88] describe the implementation of a 4D animation system. They use the technique of employing the hardware matrix multiplier on their graphics computer to do the 4D rotation and scaling operations. Most of the paper, however, is concerned with rendering issues. Much more complete is Banks' [Ban92, Ban93] doctoral dissertation on the subject of interaction and display of surfaces in 4D. His thesis contains a wealth of information touching upon several important areas. He examines the use of multiple joysticks, spaceballs, and other exotic input devices for mapping the user space into the 4D world coordinates for the purpose of performing rotations and translations. Banks provides a thorough study of which mappings promote the best kinesthetic sympathy with the user. Like Banchoff [Ban86], Banks also uses ribboning as an effective visualization tool. With ribboning the surface is either entirely transparent or entirely opaque, so it works well with current graphics hardware, which cannot support real time rendering of partially transparent surfaces. Particularly interesting is Banks discussion of silhouettes and intersection curves. He describes innovative techniques for using the z-buffer to automatically compute intersection curves after a surface is projected to 3D.

19

Chapter 3

Knot Theory

This chapter is an overview of the knot theory relevant to issues of interest in this thesis. To learn more about knot theory, one might consult several excellent treatments of the subject [Fox62, Rol76, BZ85, Kau87, Kau91, Mur96, Kaw96] that discuss the topic in great detail, as well as popularized accounts intended for a non-mathematical audience [Neu79, Pet88, Jon90]. Recent books by Livingston [Liv93] and Adams [Ada94] are thorough reviews at an intermediate level. An excellent history of knot tabulations can be found in a paper by Thistlethwaite [Thi85]. That paper, together with papers by Millett [Mil92] and Lomonaco [Lom96], provides a good historical account of knot theory.

3.1   Introduction to knot theory

Knot theory is a branch of topology where one studies what is known as the *placement problem*, or the embedding of one topological space into another [Fox62, Lom96]. The simplest (non-trivial) form of knot theory is concerned with the embedding of one or more disjoint circles into three-dimensional space. For the purposes of this thesis a *knot* is defined to be such an embedding of a circle (*i.e.* a simple closed curve) in three-dimensional Euclidean space $\mathcal{R}^3$. Note that a mathematical knot is somewhat different from the usual idea of a knot, which can be thought of as a piece of string with free ends. The knots studied in knot theory are (almost)

<div align="center">

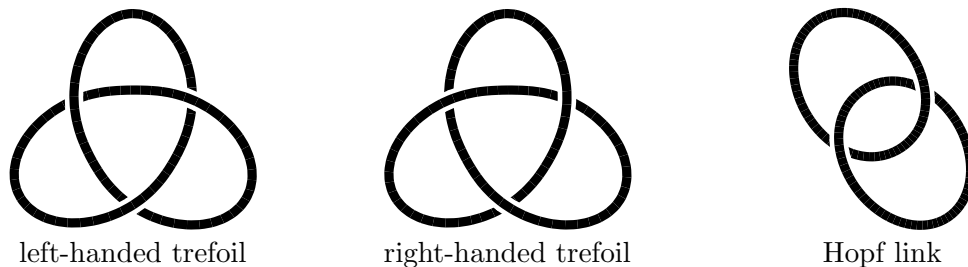left-handed trefoil   right-handed trefoil   Hopf link

Figure 3.1: The simplest knot (in two forms) and the simplest link.

</div>

always considered to be closed loops. Two or more disjoint knots together are called a *link*. Each constituent knot of the link is known as a *component* of the link. To simplify the terminology in the thesis, we will often refer to links as knots and, following Conway [Con70], use the term *proper knot* when specifically referring to one-component knots.

Two knots or links are *equivalent* if there exists an orientation-preserving *homeomorphism* on $\mathcal{R}^3$ that maps one knot onto the other.[1] A homeomorphism is a continuous one-to-one mapping with a continuous inverse. This condition is equivalent to saying that one knot may be smoothly deformed into the other.[2] Cutting the knot or allowing it to pass through itself are not permitted. In general it is a difficult problem to decide if two given knots are equivalent, and much of knot theory is devoted to developing techniques to aid in answering this question. Knots that are homeomorphic to polygonal paths in three-dimensional space are called *tame*. Other knots are known as *wild*. Most of knot theory concerns only tame knots, and these are the only knots examined here. Knots that are homeomorphic to a planar polygon are considered to be *unknotted*, or trivial.

The simplest non-trivial knot is the trefoil or clover-leaf knot. It comes in a left-handed and a right-handed form (Figure 3.1). It is not too difficult to see (but slightly more difficult to prove) that the trefoil is not equivalent to the unknot. The two versions of the trefoil are *not* equivalent under our definition of equivalence.[3] In physical terms, the inequivalence of the unknot and the two trefoils means that there would be no way to deform a piece of rope tied

---

[1] Note that this homeomorphism is a mapping from the ambient space into itself that "carries" the knot along with it. Any two knots are obviously homeomorphic if the mapping is between the knots themselves.

[2] Technically, this second form of equivalence is known as *ambient isotopy.*

[3] The two trefoils are equivalent if the homeomorphism is combined with a reflection.

in one configuration into either of the other configurations. Also shown in Figure 3.1 is the simplest non-trivial link, the Hopf link, which is just two interlocked rings, each independently unknotted.

One point worth noting is that although this thesis is concerned only with *piecewise linear* knots, all the knots presented so far appear to be *smooth* (*i.e.* differentiable) curves. The study of knots, and topology in general, can be undertaken in several different *categories.* In the *piecewise linear category*, or PL, all objects considered are piecewise linear, as are the mappings between objects. In the *smooth category*, or DIFF, everything is required to be differentiable instead. The *general category*, or TOP, drops both of these restrictions. TOP is the realm of wild knots, which we do not consider in this thesis. It has been shown that PL knot theory and DIFF knot theory are equivalent in three dimensions [Fox62], so our drawing PL knots as smooth curves is not unreasonable. Also, since it is the convention in the literature of PL topology for illustrations to convey the impression of smoothness, we will frequently shift between the two viewpoints, and will normally draw knots as looking smooth. However, the reader should bear in mind that at all times the knots considered here are actually PL knots. Curiously, in higher dimensions the PL and DIFF categories diverge [Hae62], however, this too is beyond the range of study in this thesis.

An alternative way to look at knot equivalence is by examining a projection of the knot onto a plane. Such a projection, called a *knot diagram* or *presentation*, can always be made so that the cross-over points have one of the two forms shown in Figure 3.2. It was shown by

Figure 3.2: Crossings permitted in planar projections.

Reidemeister as early as 1935 [Rei35, BZ85] that two knots are equivalent if and only if the projection of one can be converted into the projection of the other through the use of just three types of moves, along with isotopies[4] of the knot diagram in the (extended) plane. The three

---

[4]Continuous deformations.
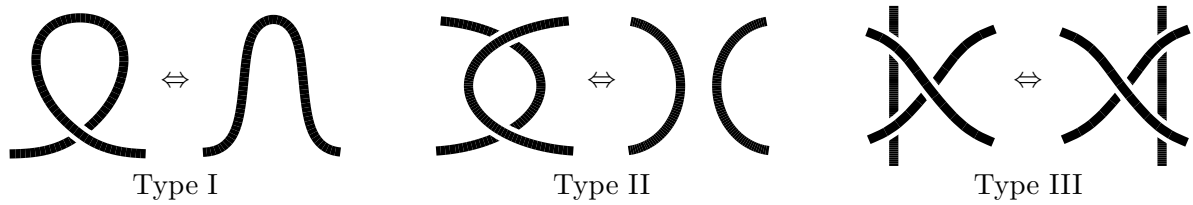
Type I   Type II   Type III

Figure 3.3: The three types of Reidemeister moves.

Reidemeister moves are pictured in Figure 3.3. Each illustration in Figure 3.3 represents a class of moves; the reader is asked to infer the remaining cases (*i.e.* using reflections and rotations).

Using the Reidemeister moves it is clear that any knot has an infinite number of distinct projections in the plane, of arbitrary complexity. For this reason, knot diagrams are often drawn in a form that has the fewest crossings of all possible projections. Such a knot diagram is called a *minimal projection* of the knot and the number of crossings in the diagram of the form shown in Figure 3.2 is called the *crossing number* of the knot. It is generally true that this minimal projection is not unique. It should be intuitively clear that the trefoils shown in Figure 3.1 as well as the unknot do have unique minimal projections, with crossing numbers of three and zero, respectively. The trefoil is the only knot with a crossing number of three. Figure 3.4 (seen before in Figure 1.2) shows examples of the unknot and the trefoil in non-minimal projections with ten crossings each and a non-minimal projection of the Hopf link with eight crossings. These diagrams provide some indication that the minimal crossing number of a knot is generally not easy to determine.
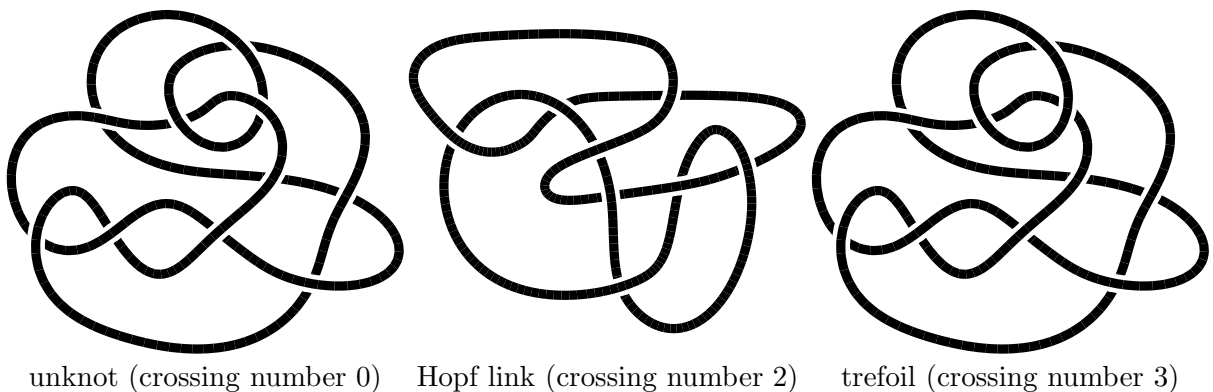

unknot (crossing number 0)   Hopf link (crossing number 2)   trefoil (crossing number 3)

Figure 3.4: Examples of non-minimal projections

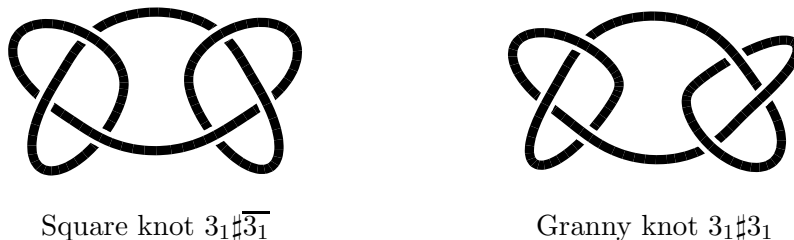Square knot $3_1 \sharp \overline{3_1}$ Granny knot $3_1 \sharp 3_1$

Figure 3.5: Examples of composite knots.

The crossing number provides a intuitive measure of complexity of a knot, and it has been the custom until recently to rank knots in catalogues according to their crossing numbers ([CDW98] take a complementary approach, see Chapter 8 for a discussion). For these purposes, it is necessary to list only *prime* knots. These are knots that cannot be subdivided into two or more simpler non-trivial knots. Knots (and links) that are not prime are called *composite*. Composite knots correspond to the operation of tying two separate knots sequentially in a piece of rope, as shown in Figure 3.5.

A catalogue of all prime knots with ten crossings or fewer (in a minimal projection) and all prime links with nine crossings or fewer, complete with diagrams, is given in Rolfsen's book *Knots and Links* [Rol76]. This catalogue, arguably the most well-known of all knot catalogues, depends on two important earlier enumerations, that of Alexander and Briggs [AB27] and another by Conway [Con70]. Alexander and Briggs produced the first complete catalogue of knots up to nine crossings in which each knot was known to be distinct from others in the catalogue. Conway extended this to ten crossings in 1970, including links with up to five components as well.[5]

Knots and links in this range have a standard naming convention (sometimes called the Alexander-Briggs-Conway-Rolfsen notation). The knots are named according to their minimal crossing number, with a subscript indicating the (mostly) arbitrary ordering among knots with the same crossing number. Because mirror images are generally ignored for purposes of enumeration in knot catalogues, there is only one knot with three crossings, namely the trefoil denoted by 3_1. The only knot of four crossings, the figure-eight knot, is 4_1 (Figure 3.6). There are two knots of five crossings, 5_1 and 5_2, three with six crossings, 6_1, 6_2, 6_3, and so on. For

---

[5]Conway also enumerated 11 crossing knots.

Number of	Crossing number								
components	2	3	4	5	6	7	8	9	10
1	0	1	1	2	3	7	21	49	165
2	1	0	1	1	3	8	16	61	184
3	0	0	0	0	3	1	10	21	73
4	0	0	0	0	0	0	3	1	21
5	0	0	0	0	0	0	0	0	3

Table 3.1: Number of prime knots and links to ten crossings.

multi-component links, a superscript indicates the number of distinct components in the link. An example is the Hopf link, 2_1^2 (see Figure 3.1), which is the only knot or link with a crossing number of two. Table 3.1 shows the number of knots and links of ten or fewer crossings up to and including Conway's results.

Recently knot enumeration has taken a huge leap forward, largely due to using computer techniques and efficient means of encoding knot diagrams. Hoste, Thistlethwaite, and Weeks have now determined that there are 1,701,935 prime knots[6] with 16 or fewer crossings [HTW98]. To check results, the researchers worked in two independent teams using completely different methods, none-the-less coming up with the same list of knots. In Chapter 6 we will return to the discussion of the work of one of these teams (that of Thistlethwaite) when we consider the issue of drawing knot diagrams for these "newly discovered" knots.

Although mathematicians are not concerned with mirror images when constructing knot catalogues, it is useful to have a notation for the mirror image of a given knot. The left and right hand trefoils shown in Figure 3.1 will be denoted as 3_1 and $\overline{3_1}$, respectively. Most knots are not equivalent to their mirror images; these knots are known as *chiral* knots. Some knots, such as the figure-eight knot (4_1) *are* equivalent to their mirror image and are called *achiral*. Chirality can be important when forming composite knots. For example, the two composite knots shown in Figure 3.5 are actually inequivalent, although this fact is not at all easy to prove.

With the figure-eight knot it is easy to demonstrate achirality; the diagrams in Figure 3.6

---

[6]The researchers included the unknot as a "knot", hence the total count of 1,701,936 in the title of the paper. It would really be better to refer to the unknot as the *trivial knot*.
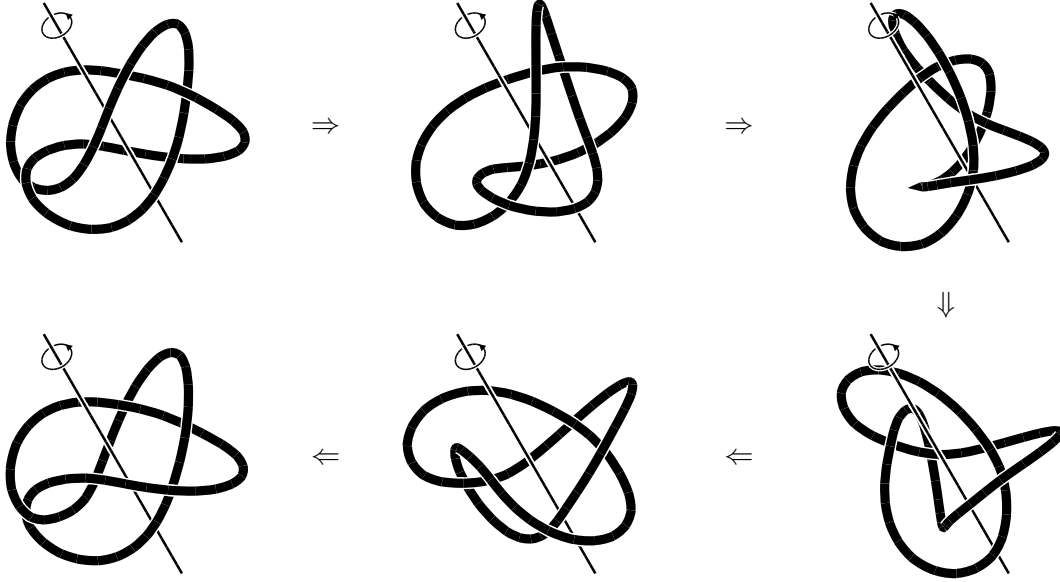
Figure 3.6: The simplest achiral knot, the figure-eight knot. Rotation about the indicated axis by 180° transforms the knot into its mirror image.

constitute a proof. It should be pointed out that the figure-eight knot illustrated exhibits *rigid geometric achirality* in that a rigid rotation of the knot by 180° is sufficient to map it into its mirror image. When we speak of knot achirality, we generally mean it in a less restrictive sense. A given embedding of an achiral knot will usually not be in a rigidly achiral position, however, it may be deformed by smooth topological transformations into its mirror image. The embedding of the figure-eight knot shown in Figure 3.6 exhibits several interesting symmetries in addition to rigid geometric achirality. Discussion of this specific case will be taken up in Chapter 8.

Some surprise was generated in the knot theory community by the previously mentioned knot enumeration work of Hoste, Thistlethwaite, and Weeks [HTW98] A long-held[7] conjecture about achiral knots with an odd crossing number, namely that none existed, was proven false with the discovery of a single counterexample with 15 crossings. This knot is shown in Figure 3.7, and doesn't look obviously achiral. Its achirality may be proven using the computer program *SnapPea* written by Weeks [Wee90, HW92] using techniques not discussed in this thesis.
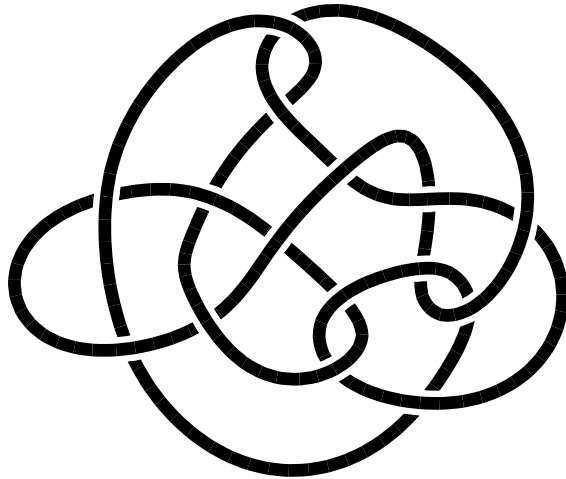
---

[7]More than a century.

Figure 3.7: The single known example of an achiral knot with an odd crossing number.

3.2 Conway's tangle calculus

In 1970 John Conway [Con70] injected new life into the theory of knots by inventing a powerful notation system that allowed the efficient enumeration of many more knots and links. Conway's system is based on the idea of *tangles*, or sections of a knot diagram consisting of two separate strings with open ends, as shown in Figure 3.8. Conway considered only two-string (four-input)

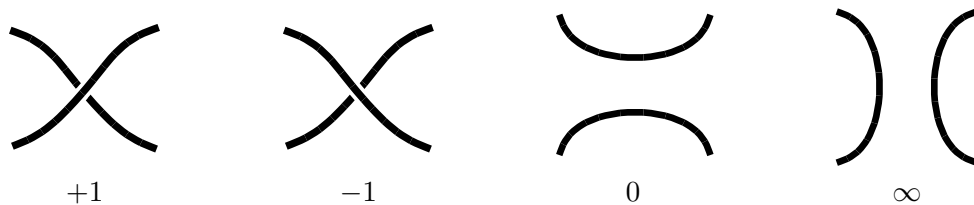|			
+1	−1	0	∞

Figure 3.8: Conway's fundamental tangles.

tangles, however, tangles with any number of strings may also be considered [APR89, Rol94].

The fundamental tangles may be combined using the algebraic operations as shown in Figure 3.9. Conway's "ⓛ" symbol indicates the orientation of the inserted tangle. These operations produce more complex tangles from simpler tangles. For example, two +1 tangles may be summed to form a +2 tangle. Iterating this process using only the +1 or −1 tangles generate the *integral tangles*, a few of which are shown in Figure 3.10. Integral tangles may be combined
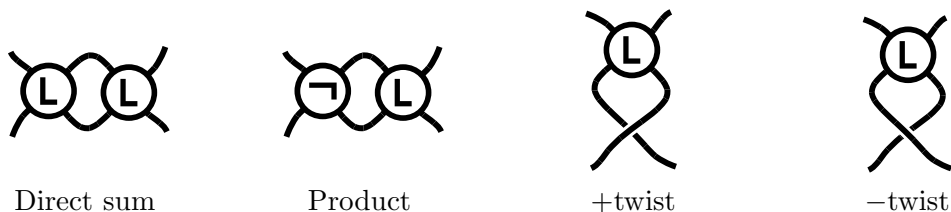
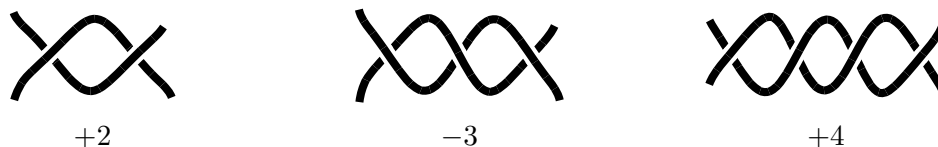Figure 3.9: Algebraic operations on tangles.

Figure 3.10: Integral tangles.

together using the product operator as shown in Figure 3.11. Note that the product operator is
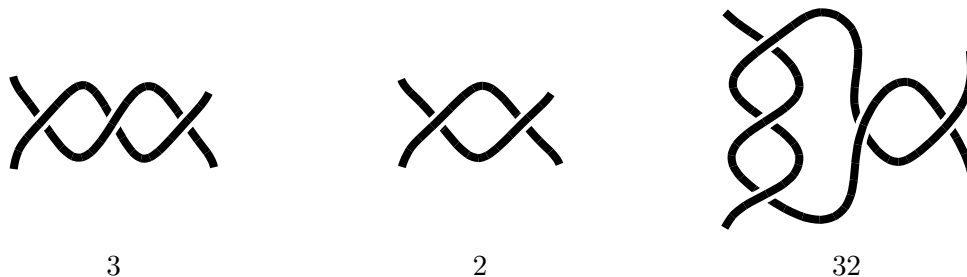

Figure 3.11: Construction of a tangle product.

similar to the sum operator except that the first multiplicand is reflected and rotated. Conway's product is left-associative, *i.e. abcd* = $(((ab)c)d)$. Figure 3.12 shows that a different tangle results if the product is taken right-associatively instead. Tangles formed in this way (left associative products of integral tangles) are known as the *rational tangles.*

Eventually, to produce a knot diagram, the free ends of the tangles must be spliced together somehow. Conway provided a number of what he termed *basic polyhedra* as templates into which one might insert various tangles. Several of these are shown in Figure 3.13, redrawn but topologically equivalent to Conway's originals. The simplest of these is the 1*-polyhedron, which joins the top two ends (NE and NW) and the bottom ends (SE and SW) to form what is also called the numerator [Kau87] of the tangle. For example, inserting a 3-tangle into the
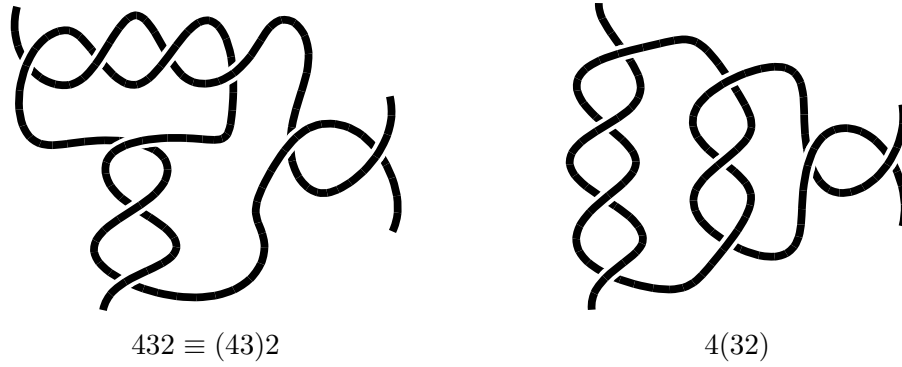
$$432 \equiv (43)2 \qquad\qquad 4(32)$$

Figure 3.12: Left associativity of tangle product.

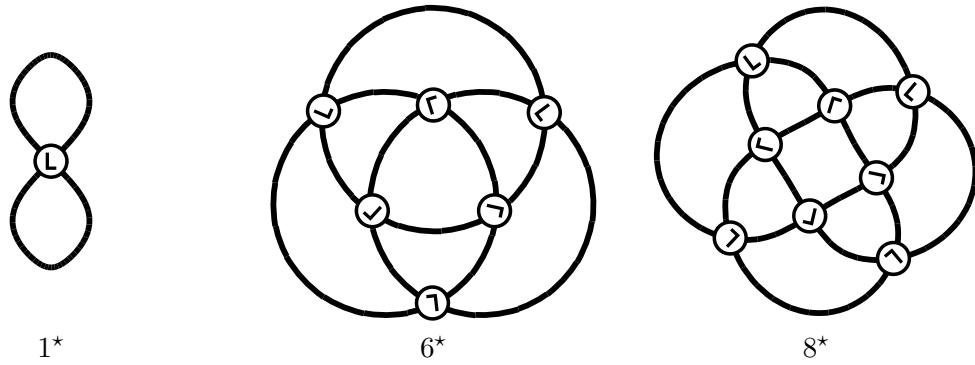$$1^\star \qquad\qquad 6^\star \qquad\qquad 8^\star$$

Figure 3.13: Several basic polyhedra.

$1^\star$-polyhedron produces the knot on the left in Figure 3.14, easily recognized as the trefoil. Conway's convention is that the notation for the knot is the same as the inserted tangle in the case that the $1^\star$-polyhedron is used, so the trefoil may be compactly denoted by the single numeral 3.

Constructing knots using basic polyhedra more complicated than $1^\star$ requires that some of the inserted tangles be reflected or rotated in a manner similar to the product operation shown in Figure 3.9. Just as in that case, the "Ⓛ" symbols guide the insertion of the tangles. Inserting a $+1$-tangle into each "slot" of the $8^\star$-polyhedron produces the knot shown on the right in Figure 3.14. Although for this case Conway's notation for the knot is simply $8^\star$, for most knots the notation is somewhat more involved. The reader should consult Conway's paper [Con70] for more detail.

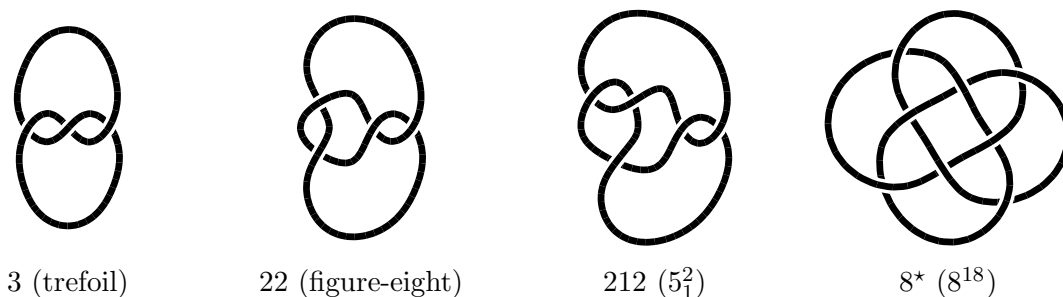| 3 (trefoil) | 22 (figure-eight) | 212 (5_1^2) | 8* (8^{18}) |

Figure 3.14: Knots with their Conway notations. The first three are constructed from the 1*
polyhedron.

This review of Conway's calculus has necessarily omitted many details and applications. In particular, the methods used by Conway in eliminating duplicates in his tables is extremely interesting. The notation system is cleverly designed to make complex operations on knot diagrams correspond to simple manipulation of the tangle strings. Also of considerable theoretical importance is a curious relationship discovered by Conway between the *rational knots* and the rational numbers (written as finite continued fractions). The rational knots are the knots obtained by inserting a rational tangle into the 1*-polyhedron. Perhaps the greatest single contribution of Conway's paper is the concept of the tangle itself. The reader is urged to consult the original paper [Con70] or more recent work that focuses on the tangle calculus [ES90] or uses tangles in innovative ways [Rol94] to learn more about these interesting applications. A precise grammar for the Conway notation may be found in the KnotPlot manual [Sch97b].

3.3 Topological invariants

Much of knot theory is concerned with providing methods for determining whether two knots given in different presentations are actually the same. So far, the method we have seen is to provide a direct proof by specifying a sequence of knot diagrams related by Reidemeister moves. Actually this is the *only* known technique for showing knot equivalence in the general case. Furthermore, no algorithm is known that will always succeed at producing a proof of knot equivalence. Even the easier problem of detecting knottedness, although decidable in
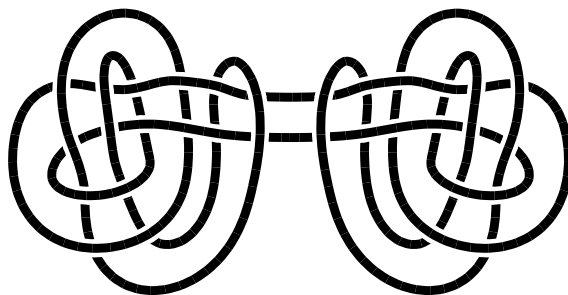
Figure 3.15: The Freedman unknot, another "difficult" unknot with no simplifying Reidemeister move.

principle [Hak61, Hem92], is only known to have an algorithm of such high complexity that it is useless for all practical purposes. Recent results of Haas, Lagarias and Pippenger [HLP97] show that the problem of detecting unknottedness is in the complexity class **NP**. This is "good news" in that, although still difficult, the problem is easier than had hitherto been believed.

The difficulty in showing knot equivalence algorithmically is that often a knot diagram must be made more complex before it can be simplified (where complexity is defined as the number of crossings in the diagram). The "monster" from Figure 1.1 as well as the "Freedman (un)knot" (taken from [FHW94]) in Figure 3.15 cannot be simplified by either one of the first two Reidemeister moves (Figure 3.3), both of which reduce the number of crossings. In fact, as will be shown in Chapter 7, the first of these unknots must go from a ten crossing projection to (at least) a twelve crossing projection before it can be simplified. Along these lines, Ochiai [Och90] has shown that there are arbitrarily complicated diagrams of the unknot that lack what he calls *n-waves*. These are specific types of underpasses (or overpasses) in the knot diagram which, if they existed in all diagrams of the unknot, would imply that an easy (*i.e.* polynomial time) algorithm could be developed for determining unknottesness (since the *n*-waves themselves are easily detected). In a similar vein, Snoeyink [Sno90] has constructed a class of complicated unknots that foil attempts at a polynomial time algorithm for detecting unknottedness using one specific approach (based on Seifert spanning surfaces).

Although the general problem of knot equivalence appears to be too difficult, topologists
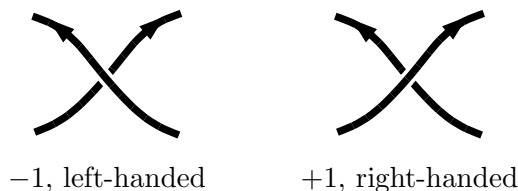
−1, left-handed   +1, right-handed

Figure 3.16: Oriented crossing types.

have developed powerful techniques that work for many cases. A *topological invariant* of a knot (or knot diagram) is a mapping from the space of knot embeddings into another space (examples are the integers, real numbers, Laurent polynomials, *etc.*) in such a way that the value of the invariant doesn't depend on the way the knot is embedded in space. With regard to a knot diagram, this is equivalent to requiring invariance under the Reidemeister moves. Using such an invariant, it is possible to show that two knots are actually distinct if they have differing values of the invariant.

A trivial example of a knot invariant is the number of components in a link. Two links with different numbers of components are obviously inequivalent; this is easy to prove by noting that the Reidemeister moves do not change the number of components. The number of components is probably the weakest knot invariant of any interest. The invariants considered in this thesis are considerably more powerful, however, a knot invariant that distinguishes all knots and links and is "not too hard" to compute still eludes knot theorists.

3.3.1 Linking number

For links with two or more components, a simple knot invariant somewhat more useful than the number of components is the *linking number*. This may be defined in a number of ways ([Rol76, page 132] gives eight equivalent definitions); the simplest in terms of a knot diagram is to consider all places in the projection where one component crosses over a different component (self-crossings are ignored). Each crossing is counted as a −1 or a +1 depending on whether the crossing is a *left-handed* or *right-handed* crossing as shown in Figure 3.16. Note that for each component an *orientation* must be chosen. The linking number $L(A, B)$ between two

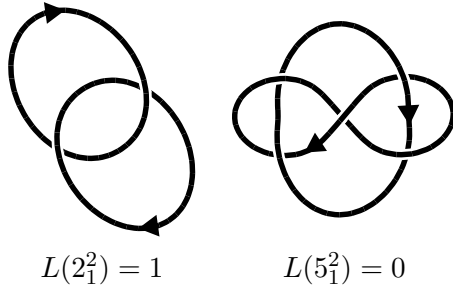$$L(2_1^2) = 1 \qquad L(5_1^2) = 0 \qquad\qquad L(6_1^2) = 3 \qquad L(6_2^2) = 3$$

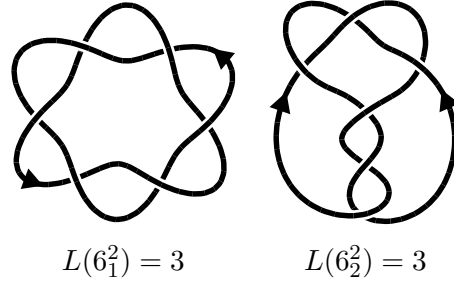Figure 3.17: Two links distinguished by linking number.

Figure 3.18: Two inequivalent links with the same linking number.

components A and B in a link is defined to be one half of the sum over all crossings of the two components.[8] Reversing the orientation on one component changes the sign of the linking number. It is not hard to show that the linking number is a knot invariant. Again, invariance under the Reidemeister moves is all that is required. Using the linking number allows one to prove that the two links shown in Figure 3.17 are different. However, Figure 3.18 shows that the linking number is not a complete invariant. In fact, because each component may be arbitrarily knotted, it's not difficult to see that there must be an infinite collection of inequivalent links that have a given linking number.

It is useful to define several concepts related to linking number that will be discussed later in the thesis. Two components in a link with a linking number of zero are said to be *homologically unlinked.* Equivalently, the two components may be unlinked by applying a smooth deformation that allows each component to pass through itself. If two components in a link, A and B, may be unlinked by having only component A pass through itself then component A is *homotopically unlinked* from component B. Note that this relation is *not* symmetric [Rol76].

Another useful definition is the *writhe* of an oriented knot diagram. The writhe, w, is defined in a similar manner to the linking number, except that a sum over *all* crossings is considered, not just crossings between two different components. Here, as with the linking number, we count a ± 1 for each crossing according to its type shown in Figure 3.16. Note that the writhe is *not* a knot invariant. It is not hard to show that Reidemeister moves of types II

---

[8]The notation $L(K)$ where K is a two-component link will be used to mean the linking number of the two components.

$8_{19}, w = 8$　　　　　$8_{19}, w = 8$　　　　　$10_{161}, w = 8$　　　　$10_{162}, w = 10$
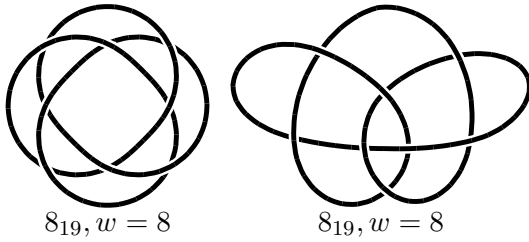
Figure 3.19: Two different minimal diagrams of the same knot with the same writhe.
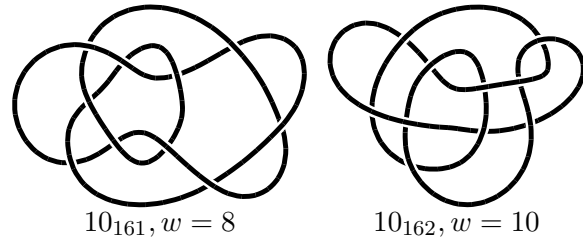
Figure 3.20: The Perko pair, two minimal crossing diagrams of the same knot with different writhes.

and III do not change the writhe, but a Reidemeister move of type I changes the writhe by ± 1, depending on the direction twisted. Properties such as the writhe that are not changed by Reidemeister moves of type II and III are sometimes called invariants of regular isotopy [Kau91].

We close this section with a cautionary tale that illustrates the subtlety of problems in knot theory. It concerns a conjecture of Tait's [Tai$\overline{76}$] on the writhe that led to the most famous case of "hidden identity" ever seen in knot theory. Tait conjectured that if a knot had two different minimal crossing diagrams, then the writhe of the diagrams would be the same. This is normally the case as can be seen from the diagrams in Figure 3.19, both minimal diagrams of the knot 8_{19}. This reasonable, but false, conjecture lead researchers to believe that the two knot diagrams shown in Figure 3.20 represented distinct knots, as they have writhes of 8 and 10 and are known to be minimal crossing diagrams. This belief persisted throughout the enumeration by Conway [Con70] (thus accounting for the different catalogue numbers of the two projections shown in Figure 3.20) until Perko showed [Per74] in 1974 that the two in fact represent the same knot. He did this by providing a simple sequence of diagrams illustrating the deformation of one diagram into the other.

3.3.2　Stick number

Another simple knot invariant is the *stick number* (or *edge number*) [Ran94, ABGW97]. This is the fewest number of straight "sticks" it takes to construct the knot. More formally, the stick number $s(K)$ of a knot K is the smallest n for which there exists an n-sided polygonal representative of that knot-type. Remarkably little is known about this apparently simple problem.

knot	sticks		knot	sticks		knot	sticks
3_1	$6^\star$		6_3	$8^\star$		7_6	9
4_1	$7^\star$		7_1	9		7_7	9
5_1	$8^\star$		7_2	9		8_1	10
5_2	$8^\star$		7_3	9		8_{19}	$8^\star$
6_1	$8^\star$		7_4	9		8_{20}	$8^\star$
6_2	$8^\star$		7_5	9		8_{21}	9

Table 3.2: Meissen's stick number results. A * indicates that the value is known to be best value possible (*i.e.* it is the stick number of the knot).

What is known is easily summarized:

- The trefoil (3_1) is the only proper knot that can constructed with six sticks, and the figure-eight knot (4_1) the only proper knot with a stick number equal to seven. The stick number for some other knots is known: Table 3.2 contains most of the known values.

- The connected sum of N trefoils has a stick number of $2N + 4$.

- For a knot K with crossing number $c(K)$, the stick number $s(K)$ is constrained by [Neg91]

$$\frac{5 + \sqrt{8c(K) + 9}}{2} \leq s(K) \leq 2c(K)$$

  The lower bound is fairly weak, since this implies that knots with ten crossings may have stick numbers as low as eight. Experimental results ([Mei96] and Chapter 8) indicate that this is unlikely.

- Rational knots have a somewhat tighter upper bound. McCabe [Ran96] has shown that for these knots $s \leq c + 4$.

- The $(m, m-1)$-torus knot (Section 5.1.2) has a stick number of $2m$ [ABGW97].

So far, most effort has gone into constructing examples that represent lower limits on the stick number. Meissen [Mei96] at the University of Iowa has pioneered much of this work. Partial results of her work are shown in Table 3.2 for purposes of comparison to research presented in
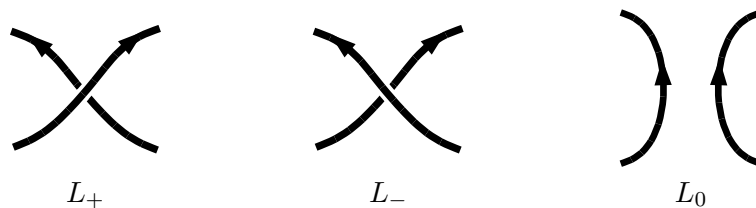
$$L_+ \qquad L_- \qquad L_0$$

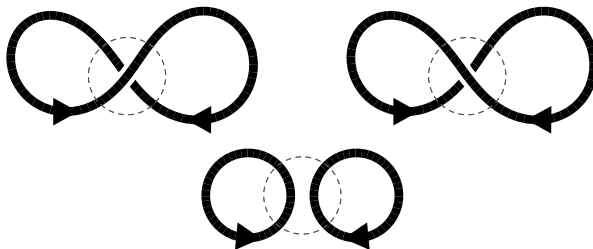Figure 3.21: Tangles appearing in skein relations.

Figure 3.22: Examples of knot diagrams related via a skein relation.

Chapter 8. Meissen arrived at her results by doing manual manipulation with a knot editing and visualization program named KED (written by Hunt [Hun96]). This work undoubtedly requires a great deal of patience, but may be the best method currently known for finding stick numbers.

3.3.3 Knot polynomials

Our examination of knot polynomials follows the excellent survey found in Wu's review paper [Wu92]. He presents a unified treatment of several knot polynomials in terms of a *skein relation*. The idea of a skein relation was invented by Conway [Con70]. It is a relation between functionals defined on a set of knot diagrams that differ only locally. Essentially, the idea is to "cookie cut" a tangle out of one knot diagram, and then to replace it with another tangle. The tangles considered are shown in Figure 3.21 and have been seen before in the discussion of the Conway notation except that here the strings in the tangle are *oriented*. One minor difference from the tangles we've seen before is that L_0 looks like Conway's ∞-tangle. Figure 3.22 shows examples of knot diagrams related via a skein relation, differing only locally within the dashed regions.

A *knot polynomial* is a mapping from a knot diagram into the Laurent polynomials[9] that is invariant under the Reidemeister moves. Alexander originally derived the polynomial that bears his name by a method completely independent of skein relations [Ale28]. The techniques he used are outside the scope of this thesis, however, consult [Rol76, BZ85] to read about the important mathematical rôle played by the Alexander polynomial (among several significant topological properties, the Alexander polynomial extends to higher dimensional knots in a way that the newer polynomials do not). Conway reformulated the Alexander polynomial using a skein relation of the form

$$\nabla_{L_+}(z) - \nabla_{L_-}(z) = z \nabla_{L_0}(z)$$
$$\nabla_{\text{unknot}}(z) = 1$$

where L_+, L_-, and L_0 represent three knot or link diagrams that differ only locally in the manner shown in Figure 3.22. The formula above specifies how the Conway polynomials $\nabla(z)$ of each of these knots are related. Calculating the polynomial for a given knot involves applying the skein relation repeatedly until one is left with a collection of unknots (or other knots whose polynomials have been previously determined). At each step, any sequence of Reidemeister moves is allowed. Such a procedure results in a binary tree, known as a *skein tree*, an example of which is shown in Figure 3.23 for the knot 6_3. The leaves of this tree are all unknots with a known polynomial $\nabla(z) = 1$. Working back toward the root of the tree while applying the skein formula allows one to compute the polynomial of the original knot.

It is not hard to show [LM88] that the calculation of the polynomial is guaranteed to terminate; in other words, the polynomial is well-defined. What is somewhat more difficult to prove is that the resulting polynomial is unique; *i.e.* it doesn't depend on the exact form of the skein tree chosen. From looking at Figure 3.23 it is probably obvious that many skein trees are possible for the same knot, since at each step there are usually many choices of which crossing the skein relation will be applied to next.[10] It is also not obvious that the polynomial is actually

---

[9]Polynomials with positive or negative exponents.

[10]The tree shown in the Figure 3.23 is unlikely to be the smallest skein tree possible for the given knot. The reader may enjoy finding a more efficient tree.
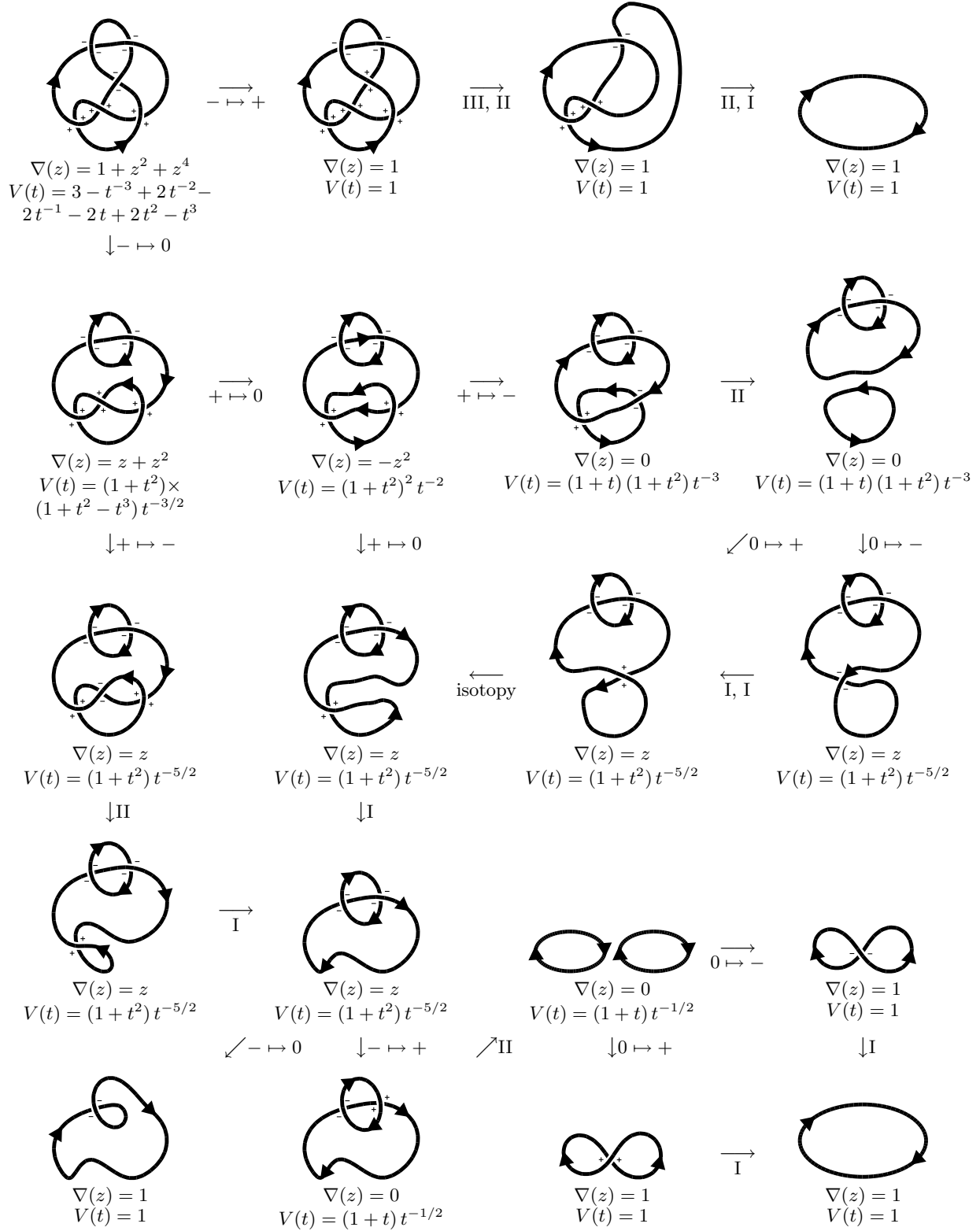
Row 1:

$\nabla(z) = 1 + z^2 + z^4$
$V(t) = 3 - t^{-3} + 2t^{-2} - 2t^{-1} - 2t + 2t^2 - t^3$

$-\mapsto+$

$\nabla(z) = 1$
$V(t) = 1$

$\overrightarrow{\text{III, II}}$

$\nabla(z) = 1$
$V(t) = 1$

$\overrightarrow{\text{II, I}}$

$\nabla(z) = 1$
$V(t) = 1$

$\downarrow - \mapsto 0$

Row 2:

$\nabla(z) = z + z^2$
$V(t) = (1 + t^2) \times (1 + t^2 - t^3)\, t^{-3/2}$

$\overrightarrow{+ \mapsto 0}$

$\nabla(z) = -z^2$
$V(t) = (1 + t^2)^2\, t^{-2}$

$\overrightarrow{+ \mapsto -}$

$\nabla(z) = 0$
$V(t) = (1 + t)(1 + t^2)\, t^{-3}$

$\overrightarrow{\text{II}}$

$\nabla(z) = 0$
$V(t) = (1 + t)(1 + t^2)\, t^{-3}$

$\downarrow + \mapsto -$ $\downarrow + \mapsto 0$ $\nearrow 0 \mapsto +$ $\downarrow 0 \mapsto -$

Row 3:

$\nabla(z) = z$
$V(t) = (1 + t^2)\, t^{-5/2}$

$\nabla(z) = z$
$V(t) = (1 + t^2)\, t^{-5/2}$

$\overleftarrow{\text{isotopy}}$

$\nabla(z) = z$
$V(t) = (1 + t^2)\, t^{-5/2}$

$\overleftarrow{\text{I, I}}$

$\nabla(z) = z$
$V(t) = (1 + t^2)\, t^{-5/2}$

$\downarrow \text{II}$ $\downarrow \text{I}$

Row 4:

$\nabla(z) = z$
$V(t) = (1 + t^2)\, t^{-5/2}$

$\overrightarrow{\text{I}}$

$\nabla(z) = z$
$V(t) = (1 + t^2)\, t^{-5/2}$

$\nabla(z) = 0$
$V(t) = (1 + t)\, t^{-1/2}$

$\overrightarrow{0 \mapsto -}$

$\nabla(z) = 1$
$V(t) = 1$

$\swarrow - \mapsto 0$ $\downarrow - \mapsto +$ $\nearrow \text{II}$ $\downarrow 0 \mapsto +$ $\downarrow \text{I}$

Row 5:

$\nabla(z) = 1$
$V(t) = 1$

$\nabla(z) = 0$
$V(t) = (1 + t)\, t^{-1/2}$

$\nabla(z) = 1$
$V(t) = 1$

$\overrightarrow{\text{I}}$

$\nabla(z) = 1$
$V(t) = 1$

Figure 3.23: Computing the Conway and Jones polynomials for the knot 6_3.

a knot invariant. In fact, the polynomial is both unique and a well-defined knot invariant; the proof is straight-forward but tedious [Kau91].

As we have said, Conway's polynomial is a reformulation of the original polynomial due to Alexander, $\Delta(t)$. The two polynomials are actually equivalent and related by

$$\nabla(z) = \Delta(t), \quad z = \frac{1}{\sqrt{t}} - \sqrt{t}$$

This difference of normalization is common in the field of knot polynomials, the same polynomial often appearing in several guises. Although the Conway polynomial is computationally equivalent to the Alexander polynomial, Conway's approach is important because it revealed that the polynomial could be calculated via a combinatorial technique on knot diagrams (the skein relations) rather than the much more abstruse topological notions used by Alexander. However, it is ironic that from Alexander's approach it is possible to derive an efficient algorithm for computing the Alexander-Conway polynomial, one that has a running time bounded by a polynomial in the number of crossings of the original knot diagram. This is in contrast to the exponential blowup we might expect occurs when computing the same quantity using a skein tree.

It turns out that the Alexander-Conway polynomial is a reasonably powerful knot invariant. Of the 249 prime proper knots with 10 or fewer crossings, 175 are distinguished by their Alexander-Conway polynomial, 68 share the polynomial with one other knot, and the remaining six knots fall into two groups of three knots with the same polynomial. The polynomial is immune to mirror images; both versions of the trefoil (Figure 3.1) have a Conway polynomial equal to $1 + z^2$. Also some simple composite knots, such as the Granny and Square knots (Figure 3.5) have the same Conway polynomial $1 + 2\,z^2 + z^4$. The reason why the Granny and the Square knots are not distinguished is related to the fact that the two trefoils have the same polynomial as well as to an interesting property about how the polynomial behaves under the direct sum of two knots. If K and L are two knots (not necessarily prime) with Conway polynomials $\nabla(K)$ and $\nabla(L)$ then the Conway polynomial of the composite knot $K \sharp L$ is given by

$$\nabla_{K \sharp L}(z) = \nabla_K(z)\nabla_L(z)$$
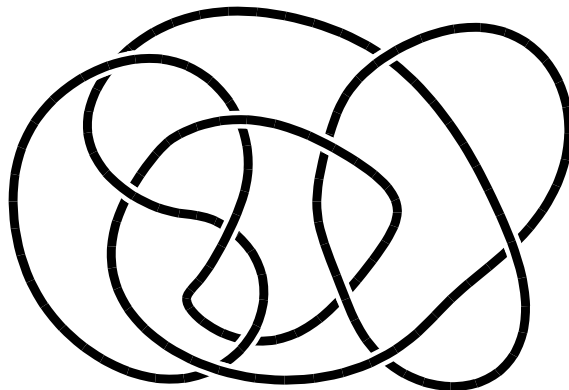
Figure 3.24: The first of an infinite series of non-trivial knots with trivial Alexander-Conway polynomial.

so we see that the failure to distinguish the Square and the Granny is the result of the failure to distinguish mirror images.

Somewhat more disheartening than the inability to distinguish mirror images is the fact that for any given knot there exists an infinite number of inequivalent knots with the same Alexander-Conway polynomial [Whi37]. In particular, there are infinitely many different knots that are indistinguishable from the unknot as far as the polynomial is concerned. The first of these, an 11-crossing knot, is shown in Figure 3.24.

For most of the history of knot theory, the Alexander polynomial was the only knot polynomial. It was quite a shock to many when Vaughan Jones [Jon85] discovered a powerful new polynomial invariant for links in 1985 defined by the skein relation

$$\frac{1}{t}V_{L_+}(t) - tV_{L_-}(t) = \left[\sqrt{t} - \frac{1}{\sqrt{t}}\right]V_{L_0}(t)$$
$$V_{\mathrm{unknot}}(t) = 1$$

The Jones polynomial is far more discriminating than the Alexander-Conway polynomial. For example, it is able to distinguish the two versions of the trefoil; the left and and right-handed trefoils having Jones polynomials $-t^{-4} + t^{-3} + t^{-1}$ and $t + t^3 - t^4$, respectively. These two polynomials illustrate a general feature of the Jones polynomial. If the Jones polynomial of a
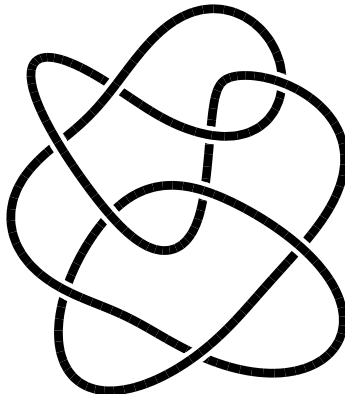
Figure 3.25: A knot, 9_{42}, with chirality not detected by the Jones polynomial

given knot is known, then the Jones polynomial of its mirror image can be found by making the substitution $t \mapsto 1/t$. In the case of the trefoil and many other knots, this property provides an easy proof of knot chirality. However, the method is not perfect; the knot 9_{42} (Figure 3.25) has Jones polynomial $-1 + t^{-3} - t^{-2} + t^{-1} + t - t^2 + t^3$ which is unchanged by the substitution $t \mapsto 1/t$ even though the knot 9_{42} *is* chiral [HW92]. The Jones polynomial shares the property of the Alexander-Conway polynomial with regard to composite knots,

$$V_{K \sharp L}(t) = V_K(t) V_L(t)$$

In this case it turns out that the Square knot with polynomial $3 - t^{-3} + t^{-2} - t^{-1} - t + t^2 - t^3$, is distinguished from both versions of the Granny knot (polynomials $t^{-8} - 2t^{-7} + t^{-6} - 2t^{-5} + 2t^{-4} + t^{-2}$ for a sum of two left-handed trefoils and $t^2 + 2\,t^4 - 2\,t^5 + t^6 - 2\,t^7 + t^8$ for a sum of two right-handed trefoils).

Jones' discovery had a profound effect on the knot theory community, causing researchers to start looking for new polynomial invariants. Remarkably, six researchers working independently in four groups almost simultaneously came up with the same polynomial (after appropriate re-normalizations) that is a generalization of both the Alexander-Conway and Jones polynomials [FYH+85]. The new polynomial, called the HOMFLY polynomial after its co-discoverers[11]

---

[11]Ruth Lawrence [Law96] makes the valid point that there was one other research team that independently discovered the same polynomial, namely that of Przytycki and Traczyk [PT87]. She suggests that the HOMFLY

<center>3,5,7,5,3   3,5,5,7,3   3,5,7,3,5</center>
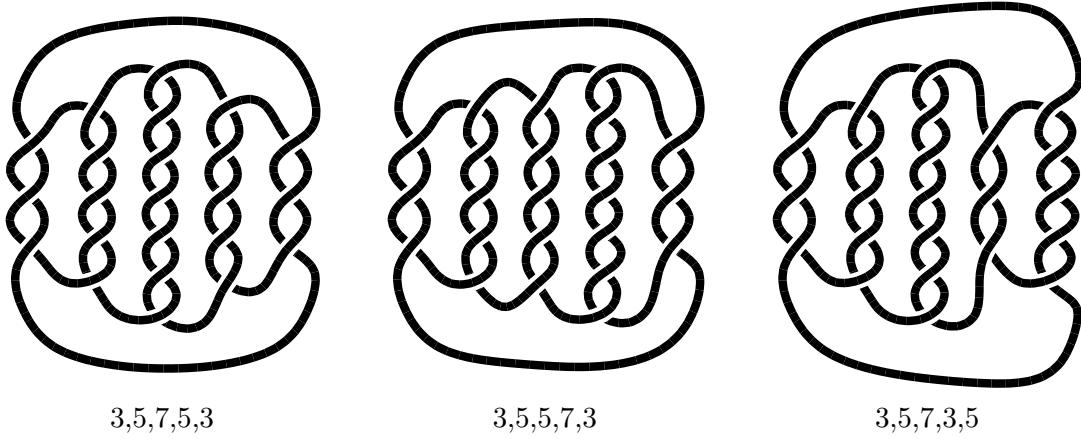
Figure 3.26: Three inequivalent "pretzel" knots (together with their Conway notation) with the same HOMFLY polynomial of $t^{-24}(-2 - 2\,t^2 + 2\,t^4 + 2\,t^6 + t^8 - z^2 - t^2\,z^2 + 2\,t^4\,z^2 + 6\,t^6\,z^2 + 12\,t^8\,z^2 + 15\,t^{10}\,z^2 + 12\,t^{12}\,z^2 + 6\,t^{14}\,z^2 + t^{16}\,z^2 + t^2\,z^4 + 5\,t^4\,z^4 + 13\,t^6\,z^4 + 23\,t^8\,z^4 + 29\,t^{10}\,z^4 + 27\,t^{12}\,z^4 + 19\,t^{14}\,z^4 + 10\,t^{16}\,z^4 + 4\,t^{18}\,z^4 + t^{20}\,z^4)$.

is defined by the relation

$$\frac{1}{t}P_{L_+}(t,z) - tP_{L_-}(t,z) = zP_{L_0}(t,z)$$

$$P_{\text{unknot}}(t,z) = 1$$

As we have said, the HOMFLY polynomial is a generalization of both the Conway and Jones polynomials, which can be obtained by making the following substitutions

$$\nabla(z) = P(1,z), \quad V(t) = P(t, \sqrt{t} - 1/\sqrt{t})$$

It is capable of distinguishing many knots that the other two polynomials do not. However, even the HOMFLY polynomial is not a complete invariant. Figure 3.26 shows three inequivalent "pretzel" knots with identical HOMFLY (and therefore Alexander-Conway and Jones) polynomials.

---

polynomial be renamed the FLYPMOTH polynomial in recognition of this fact. In addition, she mentions that this name makes a satisfying allusion to Tait's "flyping" operation. It could be added that the name FLYPMOTH is appropriate in another sense, in that it contains two insect names, corresponding to the two variables of the polynomial.

There are other knot polynomials in addition to the three discussed here, in particular the Kauffman polynomial and the Akutsu-Wadati polynomial. These polynomials are derived in a somewhat different manner from the ones we have seen, as they use a modified version of the skein tree. The Kauffman polynomial appears to be roughly equivalent to the HOMFLY polynomial for knot recognition; it is capable of distinguishing the knots shown in Figure 3.26. On the other hand there exist knots that the HOMFLY polynomial can distinguish but the Kauffman polynomial can not. Again, Wu's paper [Wu92] and Kauffman's book [Kau91] are good references.

3.4 Braid theory

Braid theory was invented in 1925 by Emil Artin [Art25, Art47] and has developed semi-independently of knot theory. For the purposes of this thesis, we will consider a braid to be a special type of knot diagram in which there are n equally spaced points connected via n strands to a second set of n points as in the examples shown in Figure 3.27. A further requirement is

$$\sigma_1^5 \qquad (\sigma_1\sigma_2^{-1})^2\sigma_2^{-1} \qquad \sigma_3^{-1}\sigma_2^2\sigma_1^{-1}\sigma_3^{-1} \qquad \sigma_1^3\sigma_4\sigma_1 \qquad \sigma_1\sigma_2^{-1}\sigma_3\sigma_4^{-1}\sigma_5$$
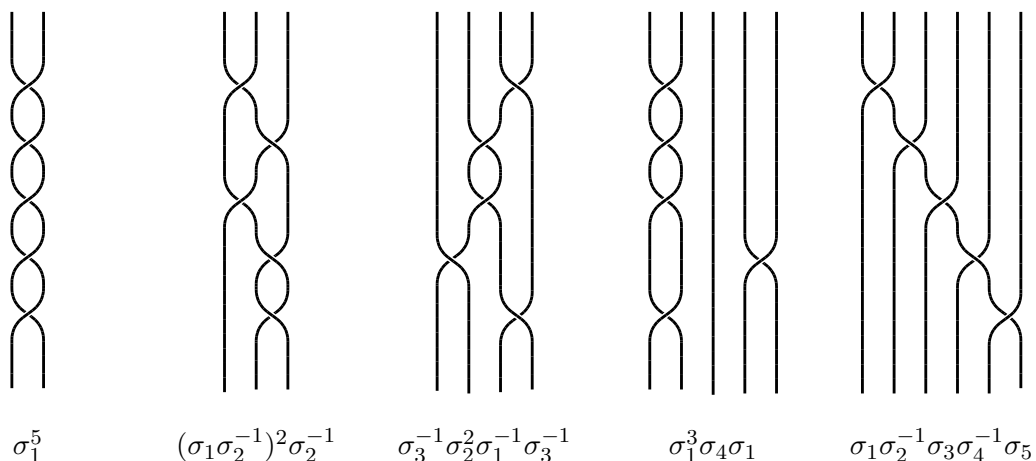
Figure 3.27: Examples of braids on 2, 3, 4, 5, and 6 strings.

that the strands proceed monotonically downward. If we draw a series of parallel lines in the plane between the two sets of points, it is not difficult to see that the braid can always be drawn such that between any pair of lines there is only one crossing of the form shown in Figure 3.28.

It is clear that any braid may be constructed by using these *generators* and the operation of
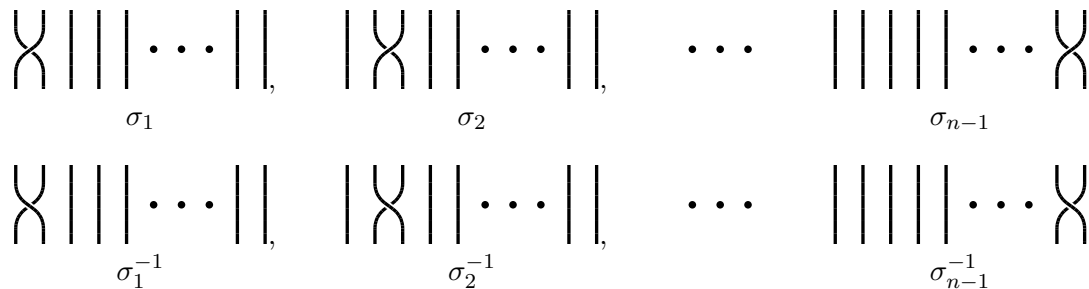
Figure 3.28: Generators of the braid group on n-strings.

composition shown in Figure 3.29. Moreover, the generators obey the simple relations

$$\sigma_i \sigma_i^{-1} = 1, \qquad\qquad i = 1, \ldots, n-1$$
$$\sigma_i \sigma_{i+1} \sigma_i = \sigma_{i+1} \sigma_i \sigma_{i+1}, \quad i = 1, \ldots, n-2$$
$$\sigma_i \sigma_j = \sigma_j \sigma_i, \qquad\qquad |i-j| > 1$$

shown graphically in Figure 3.30 (for a few specific cases). These relations are easily derived consequences of the Reidemeister moves. Together, the generators and relations form the *braid group* $\mathcal{B}_n$ on n-strings. These groups have many fascinating properties.

One further connection of braid theory with knot theory comes from a theorem due to Alexander [Ale23] that *any knot or link may be represented as a closed braid.* A braid may be closed by connecting the top ends to the bottom ends, in sequence, in such a way so as not to
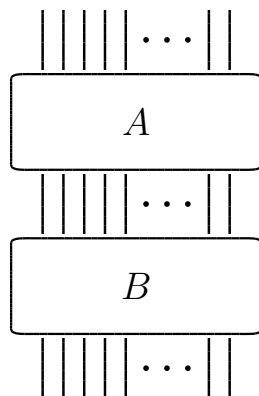
Figure 3.29: Composition of braids A and B.

44
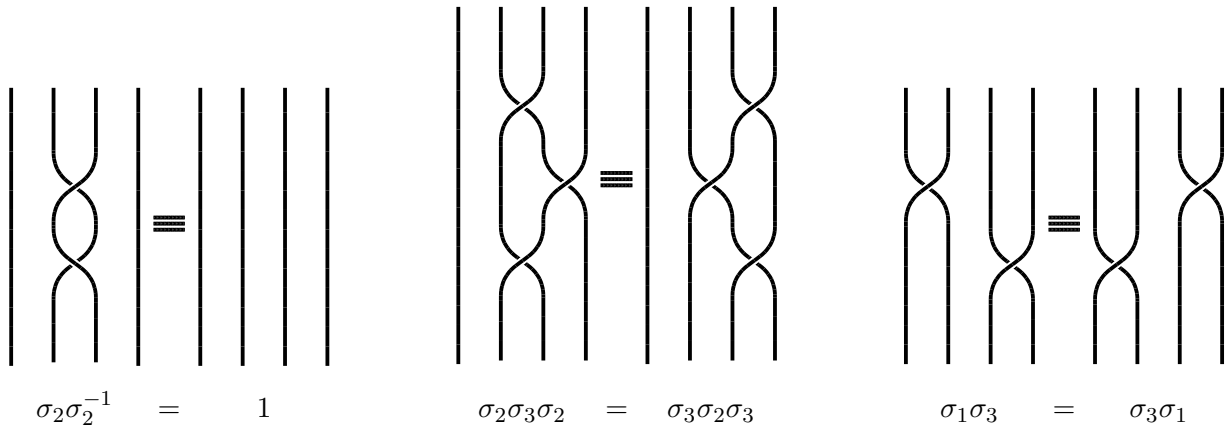
Figure 3.30: Relations in the braid group $\mathcal{B}_4$.

introduce any new crossings. An example of constructing a closed braid from a typical knot is shown in Figure 3.31. Note that in this case it is necessary to rearrange the knot diagram first so that the strings always circulate in the same direction. The essence of Alexander's theorem lies in showing that this is always possible. After the knot diagram is placed in this special form, it is easy to "read off" the braid word. This theorem has deep and important consequences in many areas [Bir74].

Our discussion of braids has been brief. Braid theory has many interesting connections with knot theory that we haven't discussed, for example in relation to the knot polynomials.[12] Also, braids have proved useful for classifying knots. Braids will appear once again in Chapter 8 with regard to the hyperbolic knot census, where they provide a convenient means to construct complicated knots. Most of the knot theory books cited at the beginning of this chapter have a section on braid theory. In particular the books by Burde and Zieschang [BZ85], Moran [Mor83], and Hansen [Han89], and papers by Birman [Bir74, Bir93] are excellent places to learn more about braids. Somewhat outside mainstream mathematics is an unusual application of braids in the theory of impossible figures [Cow74, Cow77].

---

[12]Jones discovered his polynomial by observing similarities between relations in the braid group and certain algebraic systems [Jon85].
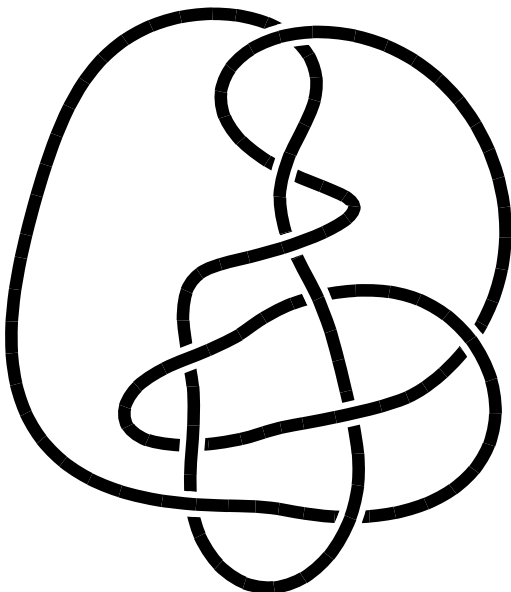
10_{102} in a minimal projection

Rearranged

σ_1^{-1}

σ_2^{-1}

σ_3

σ_1

σ_3

σ_2^{-1}

σ_1^{-1}

σ_3

$\uparrow$
start here

σ_3

σ_2^{-1}

σ_2^{-1}
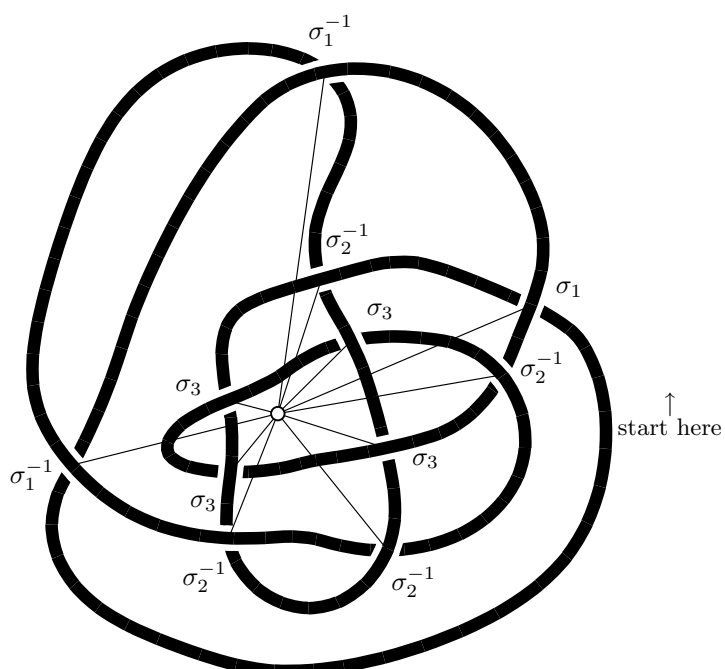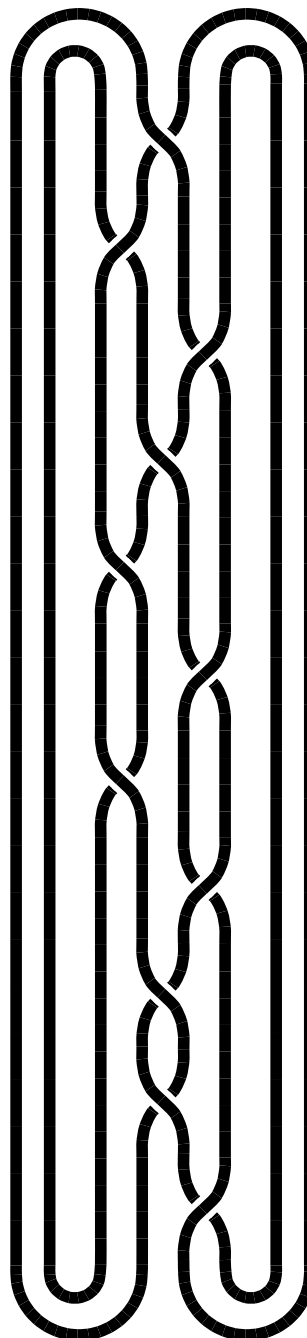
$\sigma_2^{-1}\sigma_1\sigma_3\sigma_2^{-1}(\sigma_1^{-1}\sigma_3)^2\sigma_2^{-2}\sigma_3$

Figure 3.31: Construction of a closed braid from a knot diagram.

46

3.5 Physical knot theory

Knot theory had its origin in physics. Carl Friedrich Gauss (1777–1855) made a brief excursion [Gau$\overline{33}$] into knot theory as part of his work on electrodynamics. Several decades later Sir William Thomson (Lord Kelvin) (1824–1907) had the idea that the discreteness of different knot types might correspond to the physical elements, and that each element might be a "knotted vortex" in the ether[13] [Tho$\overline{67}$, Tho$\overline{69}$, Tho$\overline{75}$]. Peter Guthrie Tait (1831–1901), a friend and colleague of Thomson, was motivated by this notion to begin the study of knots from a mathematical perspective. This resulted in three remarkable papers [Tai$\overline{76}$, Tai$\overline{84}$, Tai$\overline{85}$] that can rightfully be considered the genesis of knot theory. Tait was also among the first great knot enumerators, along with C. N. Little [Lit$\overline{85}$, Lit$\overline{89}$, Lit$\overline{90}$, Lit00] and the Rev. Thomas P. Kirkman [Kir$\overline{84}$, Kir$\overline{85}$]. Thistlethwaite [Thi85] provides an excellent discussion of this exciting period in knot theory.

The theory of relativity and the disappearance of the ether evaporated all hope that knots might have anything to do with the structure of atoms. Knot theory was then taken over by the mathematicians and it developed along mainly algebraic lines. It remained this way until Vaughan Jones' [Jon85, Jon90] discovery of a new knot polynomial in 1985 initiated a whole new era in knot theory, often referred to as a "renaissance". Jones discovered his polynomial while studying algebras related to statistical mechanics, a field apparently unrelated to knot theory. Since Jones' discovery a new field of study, *physical knot theory*, has emerged. In a remarkable turn of events it seems that knot theory has much to teach physics, and conversely the physics of knots has proved equally fascinating. The book by Louis Kauffman [Kau91] and the references it contains will give the reader an indication of the vastness of this field. For our purposes, we will be mainly concerned with areas where topological drawing can be of use in physical knot theory. The following are a sample of some of the topics that will be of interest in later chapters.

---

[13]Prior to Michelson and Morley [MM$\overline{87}$] providing strong evidence for its non-existence the ether was generally believed to be an extremely tenuous fluid that permeated the universe and acted as the medium in which electromagnetic waves were carried.

3.5.1 Energy models for knots

Part of the motivation for studying knot energies is the desire to find an optimal form for a given knot type. Most people would agree that the optimal form for the unknot is a round circle. For the trefoil, it is generally agreed that the forms shown in Figure 3.1 closely approximate the ideal[14] "Platonic" trefoil. Beyond these cases, with a few possible exceptions, there is not general agreement in the knot theory community on what constitutes an "ideal knot", despite claims to the contrary [KBM$^+$96]. At a meeting of the American Mathematical Society at the University of Iowa in 1996, Jun O'Hara outlined several desirable features that any definition of "ideal shape" should obey. The first two of these have already been mentioned. One further property is that the ideal shape for each knot type should be *unique*; if this is not the case, then there should be at most finitely many different ideal shapes for a given knot type.

It is with this quest for the ideal shape that knot energies come into play. To every embedding of a knot, a specific value for the knot energy can be given. The ideal shape would then correspond to the minimal value for this knot energy over all possible configurations for the knot. It is also desirable that if a knot is given in a high-energy non-ideal configuration, then a smooth deformation with a continuous change in the energy can bring it to the minimal energy state. To prevent self-intersections that might change the knot type, all the knot energy models studied in the literature include a term that causes the energy to go to infinity as the knot approaches self-intersection. Although something akin to knot energies have been studied in the literature for a long time, for example Milnor's work [Mil50] on the total curvature of knots, it wasn't until Fukuhara [Fuk88] introduced an energy for polygonal knots based on electrostatic repulsion that knot energies with the "blowup" property were considered. O'Hara's papers [O'H91, O'H92] placed the entire theory on a sound theoretical basis with a new energy for smooth curves.

Of the several energy models being studied in the literature, some are applicable to the smoothly embedded curves while others are appropriate for polygonal curves. As pointed out by

---

[14]In this section we will deviate from standard mathematical terminology and use "ideal" instead of "canonical"; this is in keeping with the usage in several recent papers.

Simon [Sim94] a careful distinction between these two cases is necessary. This is because energy models for the smooth case typically have a term that depends on curvature. Polygonal curves have an infinite curvature at the vertices, causing the energy models based on curvature to blow up for the polygonal case.

In this thesis we will be concerned with only a few of the knot energies that have been proposed. The most useful for our purposes is the *minimum distance* or MD energy model defined by Simon [Sim94]. Let a knot K be defined by a n-sided polygonal curve in $\mathcal{R}^3$ with vertex positions $\mathbf{v}_0, \mathbf{v}_1, \ldots \mathbf{v}_{n-1}$, and let $\mathbf{e}_k$ be the edge of the polygon extending from $\mathbf{v}_k$ to $\mathbf{v}_{k+1}$ (all numbers being taken modulo n). If $\mathrm{MD}(\mathbf{e}_i, \mathbf{e}_j)$ is the minimum distance between any point on edge $\mathbf{e}_i$ and any point on edge $\mathbf{e}_j$, then the minimum distance energy of the knot K is defined to be

$$E_{\mathrm{MD}}(K) = \sum_{\substack{\mathbf{e}_i, \mathbf{e}_j \text{ not} \\ \text{adjacent}}} \frac{\|\mathbf{e}_i\| \, \|\mathbf{e}_j\|}{\mathrm{MD}(\mathbf{e}_i, \mathbf{e}_j)^2} \tag{3.1}$$

This energy functional has a number of interesting properties. First of all, it is obviously scale and orientation independent. Equally important and more difficult to show is that there are only a *finite number of knot types with energies below a given value* [Sim94].

The second energy model we will consider is the *symmetric energy* proposed by Buck and Orloff [BO93, BO95]. It has a concrete physical interpretation that views the knot as a *radiating tube* of small radius. The symmetric energy is a measure of the amount of self-illumination of the knot. A tightly bundled knot in a non-minimal configuration will have a relatively larger self-illumination (and therefore higher energy) than the same knot in a loose, near "ideal" configuration (where most of the radiation escapes to infinity). Let K be a smooth knot parameterized by $x(t)$ and let x, y be arbitrary points $x(t), y(t)$ in K. With these definitions, the symmetric energy can be written as [BSS98b]

$$E_S(K) = \int \int \frac{|d\mathbf{x} \times r||d\mathbf{y} \times r|}{|x - y|^2} \tag{3.2}$$

where $d\mathbf{x} = \dot{x}(t)dt$ is the line element at x and $r = (x - y)/|(x - y)|$ is the unit vector in the direction of x from y. Like the MD energy, the symmetric energy is scale and orientation independent, and goes to infinity for self-intersecting curves.

The symmetric energy has an interesting relationship with the *average crossing number* of a knot. The average crossing number of a knot is the number of crossings in a projections of the knot averaged over all viewing directions.[15] Note that the average crossing number is dependent on the specific embedding of the knot and is always greater than the crossing number (which is an invariant of the knot-type). With the notation used for the symmetric energy, we can write the average crossing number of K, $\mathcal{ACN}(K)$, as [KS94, BS97]

$$\mathcal{ACN}(K) = \frac{1}{4\pi} \iint_{K \times K} \frac{|r \cdot d\mathbf{x} \times d\mathbf{y}|}{|x - y|^2} \tag{3.3}$$

For any knot K, the average crossing number provides a lower bound on the symmetric energy [BS97, BSS98b]

$$E_S(K) \geq 4\pi \mathcal{ACN}(K) \tag{3.4}$$

Since $\mathcal{ACN}(K)$ is always greater than the crossing number of K, and given that there are only a finite number of knot-types of a given crossing number or lower, this equation implies that below any specific value of $E_S(K)$, there are only a finite number of knot-types that can realize K. Also, it implies that any knot with $E_S < 12\pi$ must be the unknot.

There are numerous other energy models for knots given in the literature. The MD energy and the symmetric energy are the only two of concern in this thesis. One other model of great importance is the *conformal energy*, also known as the *Möbius energy* or the *O'Hara energy* after its discoverer [O'H91]. From a purely mathematical point of view, as opposed to the physical approach generally taken in this thesis, it is possibly the most significant of all knot functionals. It has many interesting properties, one of which is invariance under *Möbius transformations*. These are the generators of the conformal group on $\mathcal{S}^3$ (essentially, inversions through two-spheres). The conformal energy has inspired much exciting work in knot theory, of which [FH91, FH92, O'H92, Nak93, FHW94, KS94] is a sample.

---

[15]At first glance, the average crossing number might appear as a candidate for a good energy functional for knots, but we can rule it out because it does not diverge for self-intersecting curves.

3.5.2 Knot relaxation

With a knot energy functional established, one may use it to relax knots from highly-tangled, high-energy states to "ideal", (hopefully) minimal energy states. Techniques for achieving such states may be roughly divided into approaches that use gradient-descent on the energy landscape, and are therefore deterministic, and those which employ non-deterministic methods such as simulated annealing [KGV84] or other methods of random perturbation. The deterministic approach is in some ways more appealing, since it would be extremely interesting to discover that gradient-descent on a suitable knot energy always results in an "ideal" conformation, without the need to resort to non-deterministic enhancements, which often appear *ad hoc*. For more insight into these issues, the reader might consult the paper by Buck and Simon [BS93] (and its sequel by Buck and Orloff [BO93]) or the literature previously cited in the discussion of the conformal energy.

In the same paper where he proposes the MD energy, Simon [Sim94] uses a random perturbation method to find knots of minimal MD energy. He obtains good results for a few simple cases (unknots and trefoils). Also of interest in the paper is the proof that the relaxation algorithm ensures that the MD energy converges to a local minimum, while maintaining knot-type. In Simon's relaxation algorithm, the knot energy is monitonically decreasing. Wu [Wu96] has extended Simon's method to occasionally take uphill moves in the energy landscape. This allows his method to better escape any local minimum that is not a global minimum. It should be pointed out that, at least for the MD energy of knots with a "sufficient" number of vertices, there is no convincing evidence that there exist such minima.[16]

Huang, Kauffman, and Grzeszczuk [HKG96, GHK97] also use the MD energy in a simulated annealing approach. In a manner similar to one technique described in this thesis, they augment a purely physical approach with extra-physical devices such as the adding and deleting of vertices (while maintaining the total number of vertices) in the polygonal knot and a "charge drop" method where several point charges are dropped at random about the knot. These charges

---

[16]Wu has discovered instances of polygonal unknots in non-global local minima for the *thickness energy*. The thickness energy is (roughly) the diameter of the thickest piece of "rope" that could be made to follow a given embedding of knot (see [BS97] or [BSS98b] for a precise definition).

repel the regions of the knot they are close to, and aid in the untangling of complicated conformations. Huang, Kauffman, and Grzeszczuk obtain excellent results, especially for a particularly difficult unknot provided by Ochiai (originally described in [Och90]). This unknot makes its appearance in this thesis at the end of Chapter 7.

Kusner and Sullivan [KS94] relax a discrete version of the conformal energy. They use Brakke's *Surface Evolver* program [Bra92] to perform conjugate gradient descent on the energy. They find this method is effective at untangling unknots such as the Freedman unknot (Figure 3.15). Kusner and Sullivan produce a table of energy-minimized knots and links to eight crossings in an appendix. Upon casual inspection, many of these appear to be in far from ideal conformations, consisting as they do of many "tight spots", *i.e.* regions where the knot seems to be almost self-intersecting. However, as the authors point out, these are artifacts of the Möbius invariance of the conformal energy, which effectively "doesn't see" these tight spots. Actually each picture in their table represents a class of embeddings of the knot in $\mathcal{R}^3$ (more precisely $\mathcal{S}^3$), all of which are related via a Möbius transformation. Somewhat pleasing is that the Borromean rings in their usual configuration (as seen on page 14 of this thesis) are related under a Möbius transformation to their alternate configuration as three mutually-perpendicular rings. Anybody searching for Platonic ideals in knot conformations should be quite satisfied that minimizing the conformal energy arrives at both of these states simultaneously, and that the two states are really one.

3.5.3 Lorenz knots

The Lorenz equations are a system of first-order differential equations

$$\frac{\mathrm{d}x}{\mathrm{d}t} = \sigma(y - x), \qquad \frac{\mathrm{d}y}{\mathrm{d}t} = x(r - z) - y, \qquad \frac{\mathrm{d}z}{\mathrm{d}t} = xy - bz$$

that were originally proposed by Lorenz [Lor63] in a model of ocean currents. These equations are a simple example of a *dynamical system* that exhibits *chaos* [Jac90]. Figure 3.32 shows the flow on $\mathcal{R}^3$ for the Lorenz equations starting at several different positions (indicated by a dot).[17]

---

[17]In Figure 3.32 the initial positions all lie on the attractor; empirical evidence indicates that the orbit quickly converges to the attractor for any initial position in $\mathcal{R}^3$ [GHS97].

$(4.871361, 3.950401, 24.298647)$

$(-2.989052, -4.755709, 14.660322)$

$(9.563312, 11.630175, 25.553707)$

$(4.492606, 7.957013, 12.053719)$

$(-14.047131, -20.540337, 26.570669)$

$(12.846601, 17.348150, 27.255182)$
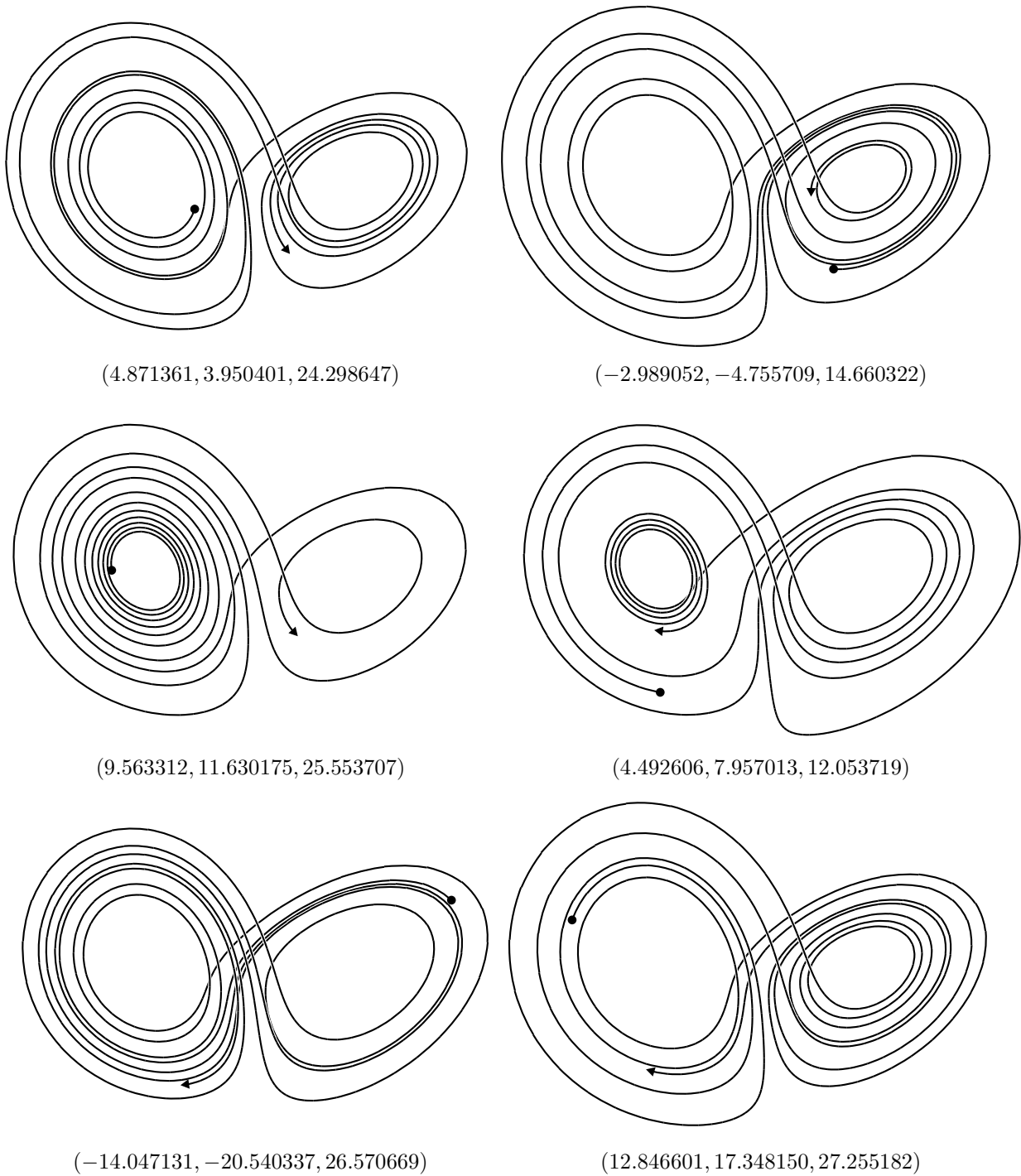
Figure 3.32: Flows on the Lorenz attractor starting from various initial values for (x, y, z) and with $\sigma = 10, b = 8/3, r = 28$.

$$(xy)^2y \qquad\qquad x^2yxy^2$$
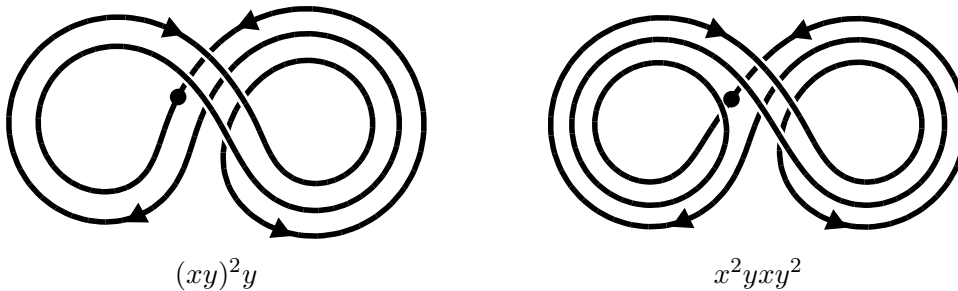
Figure 3.33: Example Lorenz knots along with their corresponding words.

It can be seen that the flow appears to be confined to a subset of $\mathcal{R}^3$. This subset, the *Lorenz attractor*, is an object of considerable complexity and interest in its own right [Jac90], however, the main concern is that there appear to exist orbits on the Lorenz attractor that are closed loops [Wil83, FW85, Hol88, Kau91, Sul93]. In fact, the closed orbits that are not unknotted are always prime knots [Wil83].

The Lorenz attractor seen in Figure 3.32 appears to be essentially two-dimensional. Orbits proceed along one of two branches of this surface, then return to a region where they again loop around one of the branches. The choice of which branch is next is chaotic. This behaviour appears to be universal for many settings of the parameters σ, b, and r, although the exact placement of the Lorenz attractor changes. Ignoring the particulars of the dynamics and concentrating on the universal behaviour allows Lorenz knots to be studied abstractly, by considering the sequence of branches taken during the orbit, as shown in Figure 3.33. Here the symbols x and y denote consecutive trips around the left and right hand branches, respectively. In this view, the Lorenz attractor is usually called the *Lorenz template* or *knot-holder*. Other knot-holders exist, more complicated than the Lorenz template. Their study has become an active area in knot-theory, and recently a book by Ghrist, Holmes, and Sullivan [GHS97] has appeared devoted to the subject. For our purposes, the Lorenz template will serve as an interesting method of constructing knots in Chapter 5.

3.6 Four dimensional knot theory

Knot theory is extendible into dimensions greater than three. The knots discussed so far are all polygons (topologically 1-spheres $\mathcal{S}^1$) embedded in Euclidean three-space, $\mathcal{R}^3$. Usually knot theorists *compactify* $\mathcal{R}^3$ by adding the point at infinity to form the three-sphere $\mathcal{S}^3$, thereby simplifying certain aspects of the theory. Viewed this way, classical knot theory studies embeddings of $\mathcal{S}^1$ into $\mathcal{S}^3$. Generalized knot theory considers embeddings of the $(n-2)$-dimensional sphere $\mathcal{S}^{n-2}$ into the n-dimensional sphere $\mathcal{S}^n$. Knotting is a *co-dimension two* phenomenon [Zee62]; all PL $\mathcal{S}^{n-2}$ are unknotted[18] in $\mathcal{S}^m$ for $m > n$. In this thesis, we will ignore the difference between $\mathcal{S}^n$ and $\mathcal{R}^n$ with regard to the ambient space. We will also restrict n to the two smallest values of interest, three and four, although diagrammatic knot theory can be extended to five and even six dimensions [Lom83, Ada94]. In a similar manner to knotting in $\mathcal{R}^3$, a knotted $\mathcal{S}^2$ in $\mathcal{R}^4$ is an embedding of the sphere (without self-intersections) that cannot be smoothly deformed into the "standard" or unknotted $\mathcal{S}^2$. It is probably clear that at least four dimensions are necessary to construct a knotted $\mathcal{S}^2$, since all smooth embeddings of $\mathcal{S}^2$ in $\mathcal{R}^3$ are topological equivalent.

There are numerous methods of constructing a knotted $\mathcal{S}^2$ in $\mathcal{R}^4$. We will consider only two different techniques, both of which construct a knot in $\mathcal{R}^4$ from a lower dimensional knotted $\mathcal{S}^1$ in $\mathcal{R}^3$. In Chapter 4 we will see another way to construct knots in $\mathcal{R}^4$. For proofs that these methods do indeed yield a knotted $\mathcal{S}^2$ and more details on the construction, refer to [Fox62, Rol76, KW77]. For terms of discussion, let (x, y, z, w) be coordinates for $\mathcal{R}^4$ and let $\mathcal{R}^3$ be the subset of $\mathcal{R}^4$ with $w = 0$.

3.6.1 Suspension

If we consider a knotted $\mathcal{S}^1$, K, lying in $\mathcal{R}^3$, we can construct a knotted $\mathcal{S}^2$ by forming the join[19] of all points on K together with each of two points on the w-axis, say $(0, 0, 0, +w)$ and $(0, 0, 0, -w)$ for any $w > 0$. That this forms a (non-self-intersecting) $\mathcal{S}^2$ is evident. That it is non-trivially knotted is not difficult to prove, but is outside the scope of this thesis as it relies

---

[18]This is not the case for the linking of spheres, see [Zee62] or [Rol76, page 7].

[19]The join of two points is the line segment joining the two points, together with the points themselves.
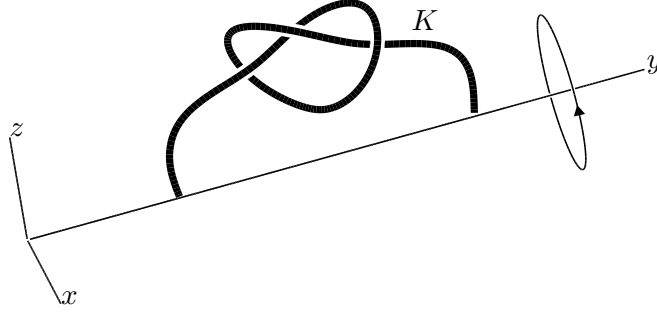
Figure 3.34: Construction of a spun knot by spinning a knot about a plane in $\mathcal{R}^4$.

on concepts developed in algebraic topology (see [Rol76] for a proof).

We will not see any pictures of suspended knots in this thesis; the manner in which they are constructed causes them to essentially "look the same" as the $\mathcal{S}^1$ that they are suspended from. Indeed, the two knots share many of the same topological properties [Rol76]. One interesting point to note is that the suspension process may be iterated, yielding higher and higher dimensional knots [Rol76]. A down side of suspension, however, is that it doesn't work for links because the suspended link is self-intersecting.

3.6.2 Spinning

Another problem with suspended knots is that they are not smooth at the poles. While this might not seem like an important fact, it is actually not possible in general to smooth out the poles [Rol76]. One way to get a smoothly embedded sphere in $\mathcal{R}^4$ is to spin a knot in $\mathcal{R}^3$ about a plane in $\mathcal{R}^4$ [Rol76, KW77]. Before considering exactly what this means, consider Figure 3.34 where a knotted open-ended arc is shown with its endpoints sitting on the y-axis. Let us first consider what it means to spin the knot about the y-axis in $\mathcal{R}^3$. If K is the knotted arc, then the spun arc can be described as

$$\mathsf{Spin}_{\mathsf{3D}}(K) = \{(r\cos\theta, y, r\sin\theta) \in \mathcal{R}^3 \, : \, (x, y, z) \in K, \ 0 \leq \theta < 2\pi, \ r = \{x^2 + y^2\}^{\frac{1}{2}}\}$$

where r is the distance in $\mathcal{R}^3$ from a point on K to the y-axis. Since K is an arc, this operation will sweep out a surface, an $\mathcal{S}^2$, in $\mathcal{R}^3$. In general the surface will be self-intersecting; this is necessarily true if K contains a knotted part as shown in the figure. However, we can do the

56

spinning in $\mathcal{R}^4$ instead, and consider the spun knot

$$\mathsf{Spin}(K) = \{(x, y, z\cos\theta, z\sin\theta) \in \mathcal{R}^4 \ : \ (x, y, z) \in K, \ 0 \le \theta < 2\pi\}$$

which produces a knotted $\mathcal{S}^2$ in $\mathcal{R}^4$, a *spun knot*, if the original arc was knotted in $\mathcal{R}^3$ [Rol76].

Spinning may be extended in a number of ways. The simplest for our purposes is *twist-spinning*, originally due to Zeeman. Referring again to Figure 3.34, imagine completely enclosing the knotted part of K by a sphere in such a way that K intersects the sphere only at the north and south poles, and that the sphere doesn't intersect the y-axis. Now during the spinning procedure, perform n complete rotations of the sphere (including the interior of the sphere) about its axis. Note that these rotations should be considered as taking place in $\mathcal{R}^3$. The resulting surface, still a non-self-intersecting $\mathcal{S}^2$, is called a *n-twist spun knot*. Remarkably, if $n = \pm 1$, then the twist spun knot is an unknot, for *any* initial knot K [Rol76]! However, the 2-twist spun knot is distinct from the ordinary (or 0-twist) spun knot. More complex deformations than mere twisting can be performed which yield still more complicated knots in $\mathcal{R}^4$. Kawauchi [Kaw96] gives a good exposition as well as the paper by Roseman [Ros89].

Chapter 4

KnotPlot

KnotPlot is a program for visualizing and interacting with 3D and 4D knots. Its features are far too numerous to describe in detail outside of a complete user's manual. Instead, this chapter will serve as an introduction to the program as well as discuss issues that are not directly related to topological drawing. Of these the most important is graphical display and user interaction with the visualization aspects of the program. Also, this chapter describes how 4D knots are constructed from 3D knots using KnotPlot.

The reader is urged to refer to the KnotPlot manual [Sch97b] for more information and detailed examples of usage. Appendix B contains information on how supplementary material related to this thesis may be obtained. This includes all experimental data described in the thesis and interactive demos, as well as the KnotPlot commands that generated most of the figures in this thesis.

In order to provide some context for discussion, Section 4.1 of this chapter contains an overview of KnotPlot. Two major topics, the creation of embeddings of knots and links and the refinement of those embeddings, are introduced briefly. These topics will be covered fully in Chapters 5 and 7 respectively. For the purposes of the remainder of this chapter, we will assume that a knot is already given and that the user is primarily interested in "visualizing" it rather than manipulating the embedding to achieve a certain result.

4.1 Overview of KnotPlot

KnotPlot is an interactive program designed to run on Silicon Graphics high-end computer graphics workstations using the GL graphics library. The program is also available for IBM RS/6000 workstations that support the GL library, and to a limited extent on Sun workstations. KnotPlot is currently installed at more than two dozen institutions and is being actively used as a research or visualization tool [KBM⁺96, MSS97, Sed96, KOP⁺97, BSS98b, CDW98]. A planned port to OpenGL [NDW93] and Java will allow the program to run on many less costly personal computers. This will greatly increase the program's availability, given the rapidly improving graphics performance of these machines and the fact that many of them already support OpenGL.

4.1.1 Creating knots and links

KnotPlot is easiest to use as simply a viewer of knots and links. To facilitate this aspect, the program has a large number of knots and links "built-in" and ready for display:

Knot catalogue — The complete Appendix C of Rolfsen's book *Knots and Links* [Rol76] is on-line (384 knots and links). Many other knots of a special nature are also included. Examples are composite knots, knots that have a special name when in a particular configuration (an example is the "Borromean rings" shown on the bottom of page 14), and ornamental knots (such as those in [Ash44] or [Bai73]). This database alone makes possible many opportunities for experimentation.

Automatic construction — Knots and links may be constructed using the tangle notation developed by Conway ([Con70] and Chapter 3). For this purpose, KnotPlot uses a *tangle calculator* that implements all of Conway's original methods and also extends the basic notation. The tangle calculator may be used either in an automatic fashion or interactively.

Special knots and links such as knot chains, torus knots (and links), and Lissajous knots [BHJS94] are also constructed automatically. A selection of "primitive" knot types, such as open straight-line knots and unknots are provided for use in other constructions.

Knots may be created from a braid word or a Dowker code (see Chapter 6) describing the knot. Chapter 8 has an example of a "real-world" application [CDW98] where this technique proved useful in constructing and visualizing rather complicated knots.

Although the above allow a user to generate and display a huge number of knots and links, there are many cases where it is necessary or desirable to create something new or different. Therefore KnotPlot allows:

Sketching — Arbitrary knots may be sketched "freehand" in 3D, marking the location of individual points along the knot (called *beads*) with the mouse and using the mouse buttons to indicate over- and under-crossings. Several modes of sketching are available, depending on the degree of exactness required.

Knot transformations — Knots may be transformed into new knots via a number of procedures. This is used to create satellite knots, doubled knots, cabled knots, and Lorenz knots.

The above methods for creating knots and links are the subject of Chapter 5.

4.1.2   Refining the embedding

After the embedding of a knot is specified, a user may use KnotPlot to refine that embedding in a number of ways:

Knot relaxation — A given knot may be *relaxed* into a smoother or simpler configuration by applying a system of forces to it. During this relaxation, collision checking is generally performed to ensure that the knot type does not change. Users can turn off collision checking to improve relaxation speed in cases where it is simple to check the knot type by inspection.

Interactive manipulation — Knots may be interactively modified by direct manipulation. This can take the form of "dragging" the knot around in three dimensions or by selectively applying forces to local regions of the knot. Other forms of interactive manipulation involve

cutting and splicing operations, duplicating and deleting components, and rotating, scaling, or translating one component with respect to another. Higher level operations such as knot mutations are also possible.

These methods for refinement are fully described in Chapter 7.

4.1.3 Computation of topological and geometric properties

The Alexander polynomial, HOMFLY polynomial, writhe, average crossing number, and thickness of knots and links can be computed using KnotPlot, as well as the Dowker code of proper knots and linking numbers of proper links. The polynomial calculations are examples of the use of KnotPlot in conjunction with other software. KnotPlot computes information about the crossings (essentially the Dowker code, extended to handle the link case as well) and passes it on to the polynomial calculation program which returns a value. For the Alexander polynomial, a small Mathematica [Wol91] program is used to compute the determinant of a matrix of polynomials. This is a needlessly wasteful step, and this author plans to implement a more efficient method in the future. The HOMFLY polynomial is computed using a program written by Robert Jenkins. The program is quite fast, however, it occasionally fails for reasons unknown. Usually a failure is obvious, but on rare occasions it is not and the program returns a wrong value. Fortunately, Jenkins provides his source code, so the problems could possibly be fixed. On the other hand, the Alexander polynomial program is much more robust, although considerably slower.

4.1.4 Open knots, knotted graphs

Two object types that are extensions to classical knot theory are also supported. The first of these are "open knots", *i.e.* what are normally considered knots by non-mathematicians, as shown in Figure 4.1. Of course, these sorts of diagrams have been seen previously in this thesis, especially in Section 3.2. Topologically, all such objects are identical, being merely embedded line-segments. However, as soon as we consider knots to be *physical objects*, as we did in Section 3.5, then the fact that such an object may be knotted locally becomes significant.
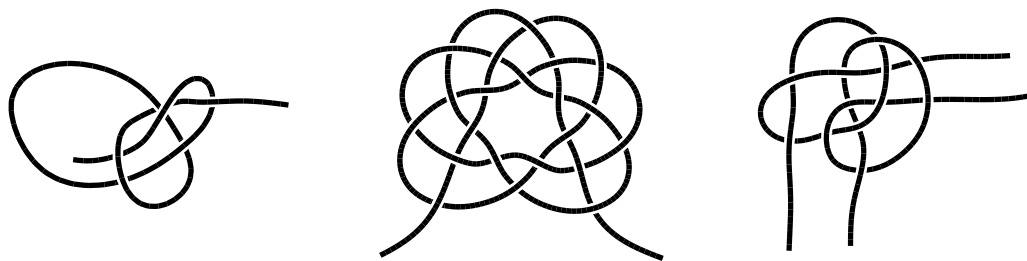
| Bowline | "Turk's head" knot | Ashley's Bend |

Figure 4.1: Open ended "everyday" knots.
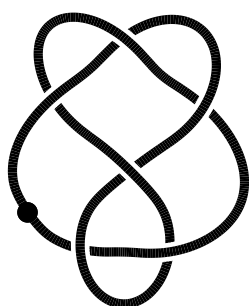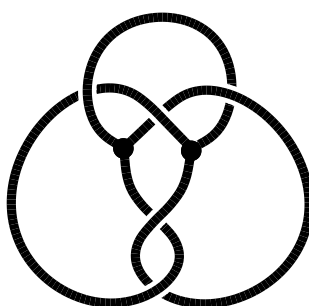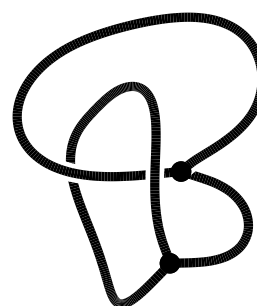
| 6_2 viewed as a graph | θ-curve | "handcuff" graph |

Figure 4.2: Knotted graphs.

The second type of objects are *knotted graphs*. The theory of knotted graphs is an extension of knot theory, classical knot theory being the "simple" case with one node and one edge, the edge being knotted as shown on the left in Figure 4.2. Conway and Gordon [CG83] initiated the theory of knotted graphs[1]. Simon's paper [Sim87] contains the beginnings of an enumeration of knotted graphs, a process that is made complicated by the large number of cases for even simple graphs such as the θ-curve shown in Figure 4.2. Much of classical knot theory can be extended to graphs; for example Kauffman [Kau89] discusses Reidemeister moves on graphs, as well as a theory of topological invariants for knotted graphs. Graphs (and open knots as well) may be relaxed just as ordinary classical knots, subject to the same collision checking algorithms.

An older form of graph representation is also implemented in KnotPlot, which proves

[1]Chemists and biologists had examined knotted graphs as models for knotted or linked molecules prior to [CG83], however, [CG83] can be considered the beginning of the knot theory of graphs.

useful in relaxing the *quad graphs* of Chapter 6. These graphs are implemented using an adjacency matrix. This is appropriate for the quad graphs as they generally have a large number of edges in proportion to the number of vertices. No collision checking is performed for these types of graphs, as none is needed for the technique used in Chapter 6. For most other purposes, this older implementation of graphs should be considered obsolete.

4.1.5 Compatibility with other systems

KnotPlot exports surface models to a number of software platforms, including ray-tracing packages such as Alias STUDIO [Ali98], and surface modelling tools including AutoCAD [Aut97] and Dragon [For97]. In addition, knot diagrams may be exported as METAFONT [Knu86] programs. The METAFONT output feature is largely unexplored, being intended for those adventurous users who might wish to create a mathematical knot typeface for use in typesetting systems such as TEX or LATEX. A simple example of this application is discussed in Chapter 8. As this thesis illustrates, PostScript pictures in numerous flavours may be generated directly using KnotPlot. Surface models exported from KnotPlot have been used in surface reconstruction studies [HDD$^+$92, HDD$^+$93, GMW97] and PostScript figures from KnotPlot have appeared in [Wu92, MSS97, CDW98].

4.2 Representation and display

Up to now the issue of exactly how knots are displayed to someone using KnotPlot has not been discussed in this thesis. KnotPlot has two major display modes. Any knot may be displayed in either a "beads and sticks" mode or in a "smooth tubes" mode as shown in Figure 4.3. These display modes correspond to the piecewise linear (PL) and smooth (differentiable or DIFF) categories in topology. As noted in Chapter 3, these categories are equivalent in $\mathcal{R}^3$. However, from an implementation standpoint the PL approach is by far the simpler of the two (this will become more apparent in Chapter 7 with the discussion on collision avoidance). Since no important aspects of knot theory are lost by adopting the PL point of view, KnotPlot uses the

| Beads and sticks mode | Smooth tubes mode |
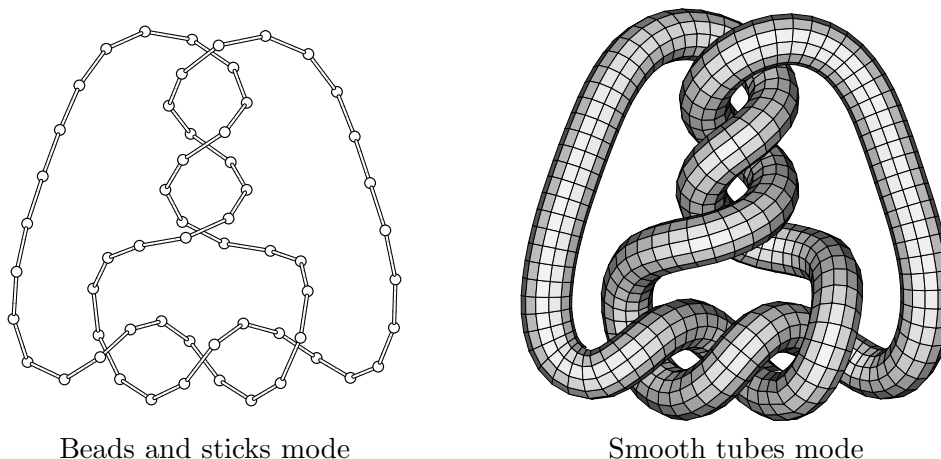
Figure 4.3: Two major display modes of the same knot. The figure on the right shows the polygonal surface constructed from the underlying PL representation shown on the left.

beads and sticks mode as its internal representation for knots. However, many people prefer the "smooth look", with the understanding that the apparently smooth curves and surfaces really represent PL objects. This is common practice in textbooks on PL topology, for example. Because of this preference, knots are generally displayed to the user in the smooth tubes mode.

It must be emphasized that the smooth surface created is merely an artifact of the underlying PL representation; it is intended *for display purposes only.* In certain situations, the smooth surface may not even be topologically "correct". For example, it may contain self-intersections if the user has selected a large value for the tube radius. The responsibility is on the user for insuring that the surface represents a valid mathematical object (although KnotPlot can provide assistance in making this judgment). The underlying PL knot, in contrast, is always guaranteed to be a valid representation of the knot type expected.

Figure 4.4 shows the two steps in going from the PL knot to the smooth surface. The first stage is to construct a smooth interpolation of the polygon. For this a simple Bézier curve [Knu86] that interpolates the midpoints of each polygonal segment is used. The control points for the Bézier curve are chosen so that it is tangent to the polygon at the midpoint of each
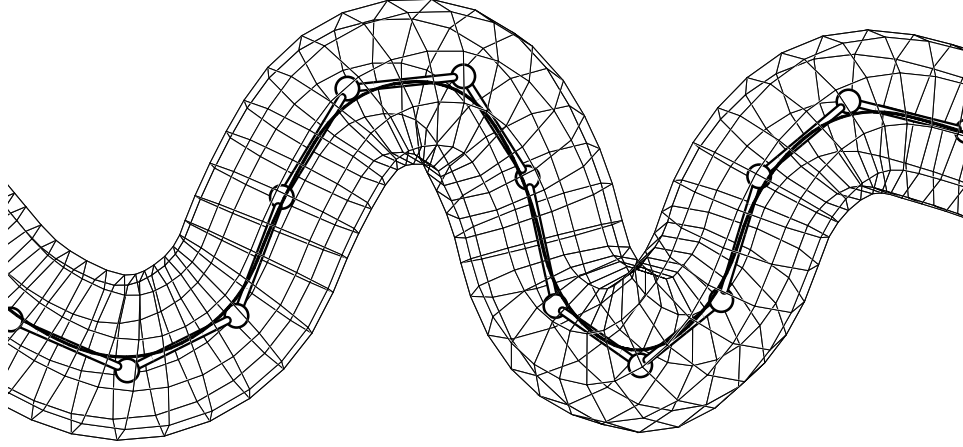
Figure 4.4: Smooth surface constructed from polygon. The solid black curve is the Bézier spline that interpolates and is tangent to the polygon at the midpoint of each edge.

edge of the polygon. For three consecutive vertex locations $\mathbf{a}$, $\mathbf{b}$, and $\mathbf{c}$, let $\mathbf{v}_1$ be the midpoint of edge $\mathbf{ab}$ and $\mathbf{v}_4$ be the midpoint of edge $\mathbf{bc}$. Then the Bézier curve between $\mathbf{v}_1$ and $\mathbf{v}_4$ is given by

$$\mathbf{p}(t) = (1-t)^3\mathbf{v}_1 + 3(1-t)^2t\mathbf{v}_2 + 3(1-t)t^2\mathbf{v}_3 + t^3\mathbf{v}_4 \ \ \text{for} \ \ 0 \le t \le 1$$

The location of points $\mathbf{v}_2$ and $\mathbf{v}_3$ is somewhat arbitrary; as long as $\mathbf{v}_2$ lies on the ray extending from $\mathbf{v}_1$ to $\mathbf{b}$, the Bézier segment will be tangent to the edge $\mathbf{ab}$ at $\mathbf{v}_1$ (similarly for $\mathbf{v}_3$). The selection

$$\mathbf{v}_2 = \frac{1}{3}\mathbf{v}_1 + \frac{2}{3}\mathbf{b} \qquad \mathbf{v}_3 = \frac{1}{3}\mathbf{v}_4 + \frac{2}{3}\mathbf{b}$$

produces good results. The curve so obtained has several desirable properties. One is the *convex hull property*; each Bézier segment is contained within the convex hull of its defining control points. This ensures that the smooth interpolant never strays "too far" from the original polygon. Another nice property is that the smooth curve created by splicing together all the Bézier segments is "mathematically close" to the original knotted polygon in that the two have many points that coincide both in location and tangent direction. One minor point is that, in general, the smooth curve will have curvature discontinuities at the endpoints of each Bézier

65
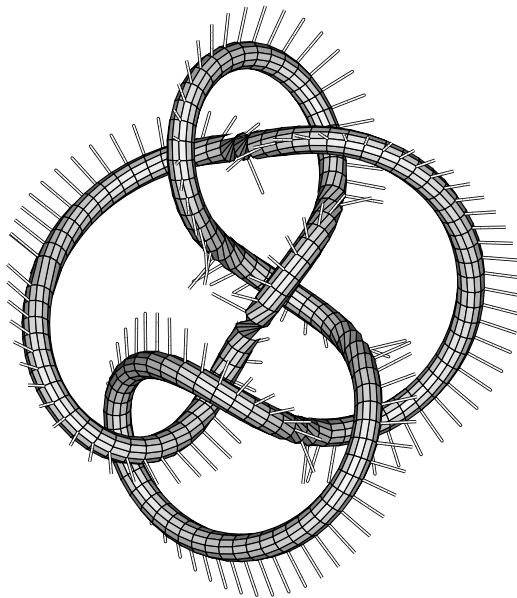
Figure 4.5: Problems resulting from using the Frenet frame.

segment. Such discontinuities are essentially impossible to see in typical situations and have no effect on any subsequent calculations.

After the first stage of finding a smooth interpolating curve, the thick tubular surface is generated. These types of surfaces, sometimes called *generalized cylinders* [SB84], are commonly occurring objects in computer graphics. Generalized cylinders are surfaces that result from extruding an arbitrary closed contour along a smooth curve, the contour forming the cross-section of the generalized cylinder. The surfaces constructed in KnotPlot are simple examples of generalized cylinders, where the contour is a regular polygon that lies in the normal plane at each point along the curve. To polygonalize the surface, it is necessary to align consecutive circular cross-sections. A first approach to doing this might define a sequence of coordinate systems based on the normal and binormal vectors of the Frenet frame (see any elementary book on differential geometry). In the case where the smooth curve has a non-zero and continuous curvature this is an acceptable solution. However, knots in KnotPlot often have several consecutive co-linear vertices. The smooth interpolant to such knots will also have segments of zero curvature where the Frenet frame is not defined. In addition, the Frenet frame may twist around the curve at inflection
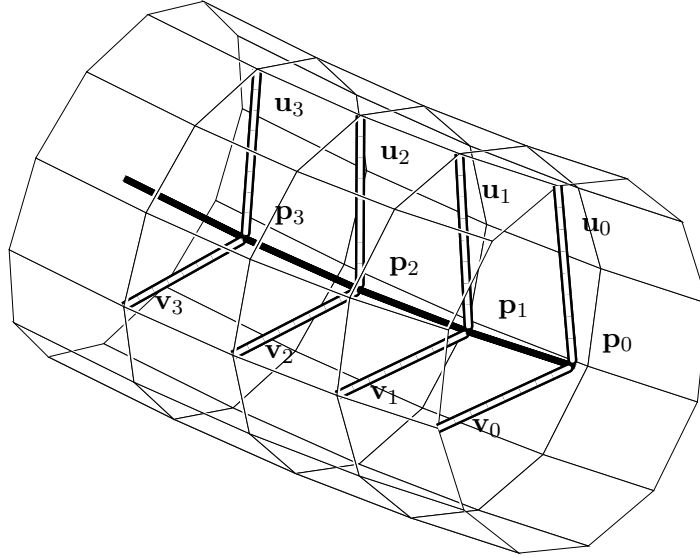
Figure 4.6: Parallel transport along the space curve.

points, making it difficult to construct the generalized cylinder easily. These difficulties with the Frenet frame are well known, as Shani and Ballard point out in their survey of generalized cylinders [SB84]. A typical knot with a tubular surface drawn based on the Frenet frame is shown in Figure 4.5. The surface shown is clearly unacceptable. In addition to gross defects in the surface due to zero points in the curvature, the surface appears to twist around excessively. These problems occur in this case even though the Frenet frame is well defined at each point along the knot.

For the reasons given above, a different method, *parallel transport,* is used instead to generate a sequence of coordinate frames along the smooth curve. To compute this sequence involves making an arbitrary initial choice for the orientation of the coordinate frame at one point on the smooth curve. Let the unit tangent vector to the curve at the point $\mathbf{p}_0$ be $\mathbf{t}_0$ and let $\mathbf{u}_0$ be an arbitrary unit vector perpendicular to $\mathbf{t}_0$. Then $\mathbf{u}_0$ and $\mathbf{v}_0 = \mathbf{u}_0 \times \mathbf{t}_0$ define the initial coordinate frame. From this frame a circular cross-section of the generalized cylinder is constructed that corresponds to the point $\mathbf{p}_0$. The coordinate frame $(\mathbf{u}_n, \mathbf{v}_n)$ corresponding to a point $\mathbf{p}_n$ further along the curve is obtained by parallel transport of $(\mathbf{u}_{n-1}, \mathbf{v}_{n-1})$ along the

67
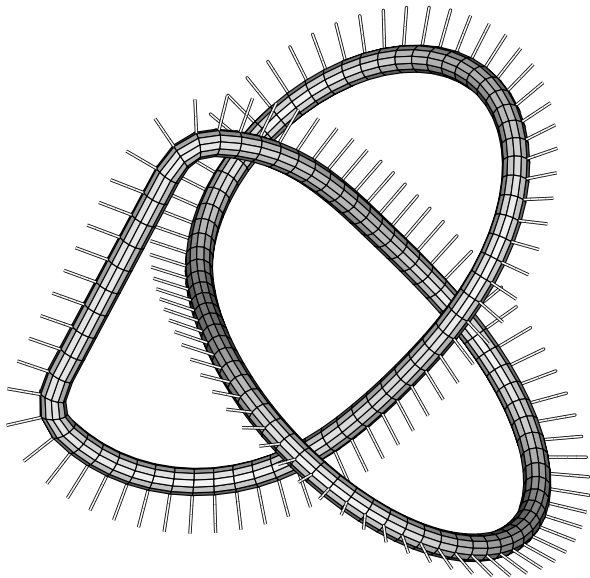
Trefoil with straight section   Smoother knot 6_3
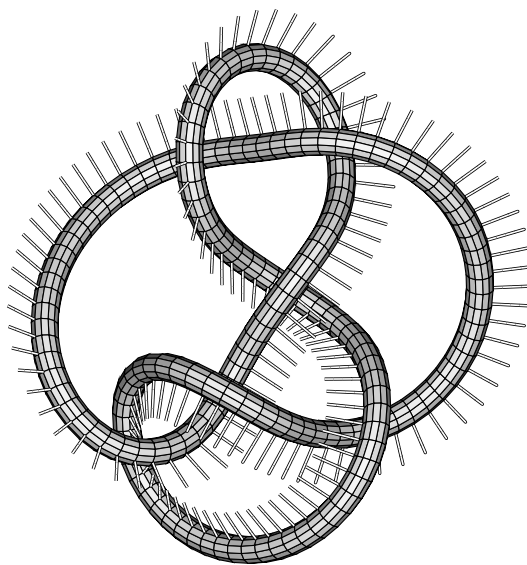
Figure 4.7: Transportation of the coordinate frame along a knot.

curve and may be computed as

$$\mathbf{u}_n = \frac{\mathbf{t}_n \times \mathbf{v}_{n-1}}{\| \mathbf{t}_n \times \mathbf{v}_{n-1} \|} \quad \mathbf{v}_n = \mathbf{u}_n \times \mathbf{t}_n$$

where $\bar{\mathbf{t}}_n$ is the unit tangent vector to the curve at the point $\mathbf{p}_n$ (see Figure 4.6). These equations are strictly valid only in the infinitesimal limit, however, in practice they produce the required results for all cases of interest in KnotPlot. Figure 4.7 shows two typical examples. The trefoil has a section of co-linear beads, an instance of a case where using the Frenet frame would lead to unpredictable results. The second example in Figure 4.7 can be directly compared Figure 4.5, where the same knot led to poor results when using the Frenet frame.

It should be noted the parallel transport is not parallel transport in Euclidean space $\mathcal{R}^3$. Rather, to understand the space in which the parallel transport takes place we have to consider the *normal bundle* associated with the space curve. The normal bundle is a type of manifold called a *fibre bundle*. It consists of a *base space*, here the knot itself, together with a collection of *fibres*. Every point in the base space has an associated fibre, in this case the normal plane at that point on the knot. This normal bundle is a perfectly good three manifold, and it can

be endowed with a natural metric from which it is possible to derive a notion of parallelism. Viewed in this way, as *parallel transport in the normal bundle*, the coordinate frames shown in Figure 4.7 seem like natural choices.

The astute reader familiar with differential geometry may protest at this point. In Figure 4.7, the vector returns to its original orientation after it has been transported through a closed path. This does not happen in general for parallel transport around a closed path in a non-euclidean manifold. It would not occur in this case either except that a small extra twist has been added between successive coordinate frames shown in Figure 4.7 so that the vector matches up with its initial orientation after being transported around the curve. This extra twist is not *ad hoc*, but rather necessary as a result of the non-euclidean nature of the normal bundle.

A final point is that our method is similar (if not identical) in effect to the *rotation minimizing frame* discussed by various other researchers [SB84, Klo86, BR91], although our derivation does not depend on that work.

4.3   Higher dimensional knots

This section deals with the construction, display, and interaction with higher dimensional knots. Of these operations, the last two lie outside the scope of topological drawing as discussed in this thesis. The construction of higher dimensional knots in KnotPlot involves topological drawing in that these knots are always constructed from ordinary knots in $\mathcal{R}^3$. After they are constructed, no topological manipulation is done with them. They should be considered rigid objects in 4D in that KnotPlot doesn't (yet) permit interaction or manipulation on them. Applying the techniques of Chapter 7 to knots in 4D offers exciting mathematical possibilities. This extension is discussed as potential future work in Chapter 9.

4.3.1 Construction

KnotPlot supports suspended and twist-spun knots via direct construction from a single 3D knot. Arbitrary surfaces in $\mathcal{R}^4$, possibly knotted or non-orientable, may be constructed with external programs and loaded into KnotPlot for viewing. All surfaces are represented as collections of quadrilaterals in $\mathcal{R}^4$, where a sufficient quantity of quadrilaterals is used so that any deviation from the intended surface (as well as the issue that the quadrilaterals might not be planar) is visually negligible.

Suspended knots

Construction of these knots is trivial. Given a knot K defined by an n-sided polygonal curve in $\mathcal{R}^3$ with vertex positions $\mathbf{v}_0, \mathbf{v}_1, \ldots \mathbf{v}_{n-1}$, the suspended knot in $\mathcal{R}^4$ is given by

$$\mathsf{Susp}(K) = \left(\bigcup_i t_i^+\right) \cup \left(\bigcup_i t_i^-\right)$$

where $t_i^{\pm}$ is the triangle in $\mathcal{R}^4$ with vertex locations $(\mathbf{v}_i, 0)$, $(\mathbf{v}_{i+1}, 0)$, $(0, 0, 0, \pm w)$ (subscripts being taken modulo n), and w an arbitrary positive real number. For display purposes these triangles are subdivided into quadrilaterals that degenerate into triangles at the "north and south poles" located at the points $(0, 0, 0, \pm w)$.

Spun and twist-spun knots

Ordinary spun-knots are a special case of twist-spun knots, for the case of zero twist. To construct a spun-knot, the user interactively positions the "spin plane" using the mouse buttons until it is situated as shown in Figure 4.8, with a knotted arc on one side of the plane and a simple arc on the other side. During this interaction, the spin plane "slides" along the knot, always intersecting the knot in at least two places. The three mouse buttons are used to modify these points of intersection as well as the angle the normal to the spin plane forms to the axis between the two points. KnotPlot ensures that spin plane intersects the knot at only two points, otherwise the spun surface will not in general be a sphere.[2] The spin plane defines a coordinate

---

[2]An interesting case is where the spin plane doesn't intersect the knot at all. In this case a spun torus results. Construction of spun tori is not possible with the current software, but is a straightforward extension.
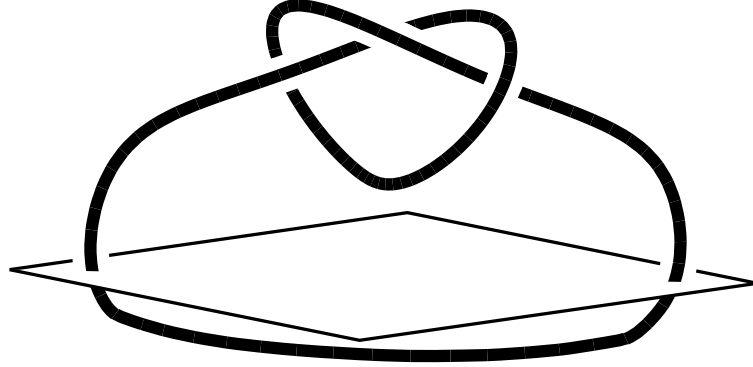
Figure 4.8: Positioning the "spin plane".

system on $\mathcal{R}^3$; the $z = 0$ plane being the spin plane itself and the z coordinate the perpendicular distance from the spin plane. Let the knot K be given as before with $\mathbf{e}_i$ the edge of K extending from $\mathbf{v}_i$ to $\mathbf{v}_{i+1}$, then the spun knot in $\mathcal{R}^4$ is the union of all spun edges of K,

$$\mathsf{Spin}(K) = \bigcup_i \mathsf{Spin}(\mathbf{e}_i)$$

The Spin of an edge $\mathbf{e}$ extending from the coordinates (x_a, y_a, z_a) to (x_b, y_b, z_b) is a collection of m quadrilaterals given by

$$\mathsf{Spin}(\mathbf{e}) = \bigcup_{s=1}^{m} q(s, \mathbf{e})$$

with $q(s, \mathbf{e})$ the quadrilateral in $\mathcal{R}^4$ specified by

$$
\begin{aligned}
q(s, \mathbf{e}) \quad = \quad & (x_a,\, y_a,\, z_a \cos(2\pi s/m),\, z_a \sin(2\pi s/m) \rightarrow \\
& (x_b,\, y_b,\, z_b \cos(2\pi s/m),\, z_b \sin(2\pi s/m) \rightarrow \\
& (x_b,\, y_b,\, z_b \cos(2\pi(s+1)/m),\, z_b \sin(2\pi(s+1)/m) \rightarrow \\
& (x_a,\, y_a,\, z_a \cos(2\pi(s+1)/m),\, z_a \sin(2\pi(s+1)/m)
\end{aligned}
$$

where the arrow indicates the cyclic ordering of the vertices. In the above, the (x, y, z) coordinates refer to coordinate frame defined by the spin plane. The value of m may be selected by the user; higher values of course produce smoother surfaces. Twist-spun knots are constructed in a similar manner, the user specifies the number of twists in addition to the location of the spin plane. We will not give the equations for the construction here, as it is straight-forward,

71

Figure 4.10: +1-twist spun trefoil. Although not apparent here, this 2-sphere is unknotted in $\mathcal{R}^4$.

Figure 4.9: Spun figure-eight knot.

being a simple rotation in $\mathcal{R}^3$ that is applied to the $\mathbf{v}_i$ as the spinning takes place (see Section 3.6.2).

4.3.2 Interaction

The interaction with higher dimensional objects in KnotPlot is fairly simple. Only global interaction (rotation) is provided. The user selects rotation angles by means of sliders. This is quite cumbersome to use and it can take a long time to find a satisfactory orientation for an object. Navigation in 4D orientation space can be expected to be rather difficult, considering its high dimensionality. It takes six angles to specify an orientation in 4D. Three of these are applied to the 4D object prior to its projection from 4D to 3D. The choice of which planes the rotations occur in is user-selectable.

4.3.3 Display

KnotPlot supports a simple display model for the visualization of higher dimensional objects. The 4D objects are projected to 3D via either orthographic or perspective projection following an initial rotation in 4D. The resulting 3D object is then rendered with standard display techniques. This usually results in a 3D object that self-intersects in the case of knotted spheres. The problems with adequately rendering such a surface have occupied much of the effort in the literature [AB90, BF88, HH91, HH92, HC93, Ban92, Ban93]. It is the opinion of this author that perhaps too much effort is expended with regard to the "proper" rendering of surfaces in 4D. Just as in ordinary 3D modelling, a wire-frame is often more descriptive of the nature of the surface than a shaded surface representation, although the latter is probably more interesting visually. Nevertheless, KnotPlot does support two modes of surface rendering. The first of these is a form of pseudo-transparency. Since real-time transparency is practically impossible in the general case on currently available graphics hardware, transparency is "faked" by exploiting features of the graphics frame buffer. The closest physical analogue to this form of rendering would be a mostly-transparent luminescent, non-refracting material — a "jelly-fish" perhaps. KnotPlot does support several different methods of wireframe rendering. The most interesting of these is where the wire frame is coloured according to the dimension that is lost in the projection from 4D. It is often easy to tell that an object such as a knotted sphere that may appear to be self-intersecting in 3D is actually not self-intersecting in 4D.

Figures 4.9 and 4.10 show wire-frame renderings of two knots (actually only one is knotted), after a perspective projection has been applied from $\mathcal{R}^4$ to $\mathcal{R}^3$. KnotPlot's other surface rendering mode uses the ribboning method described in Section 2.5. Figure 4.11 shows an example of a $+2$-twist spun knot, starting from a trefoil in $\mathcal{R}^3$.

Finally, it is possible to view a 4D knot in KnotPlot using the method of *movies*. This is beautifully illustrated in [Fox62, FR86, Lom83] The idea is to consider a sequence (movie) of 3D slices through the 4D knot. If the knot is in general position in $\mathcal{R}^4$ then each of these slices will be a union of circles (perhaps knotted or self-intersecting) in $\mathcal{R}^3$. This is analogous to the situation of having an embedded sphere in $\mathcal{R}^3$. If one takes a sequence of 2D slices
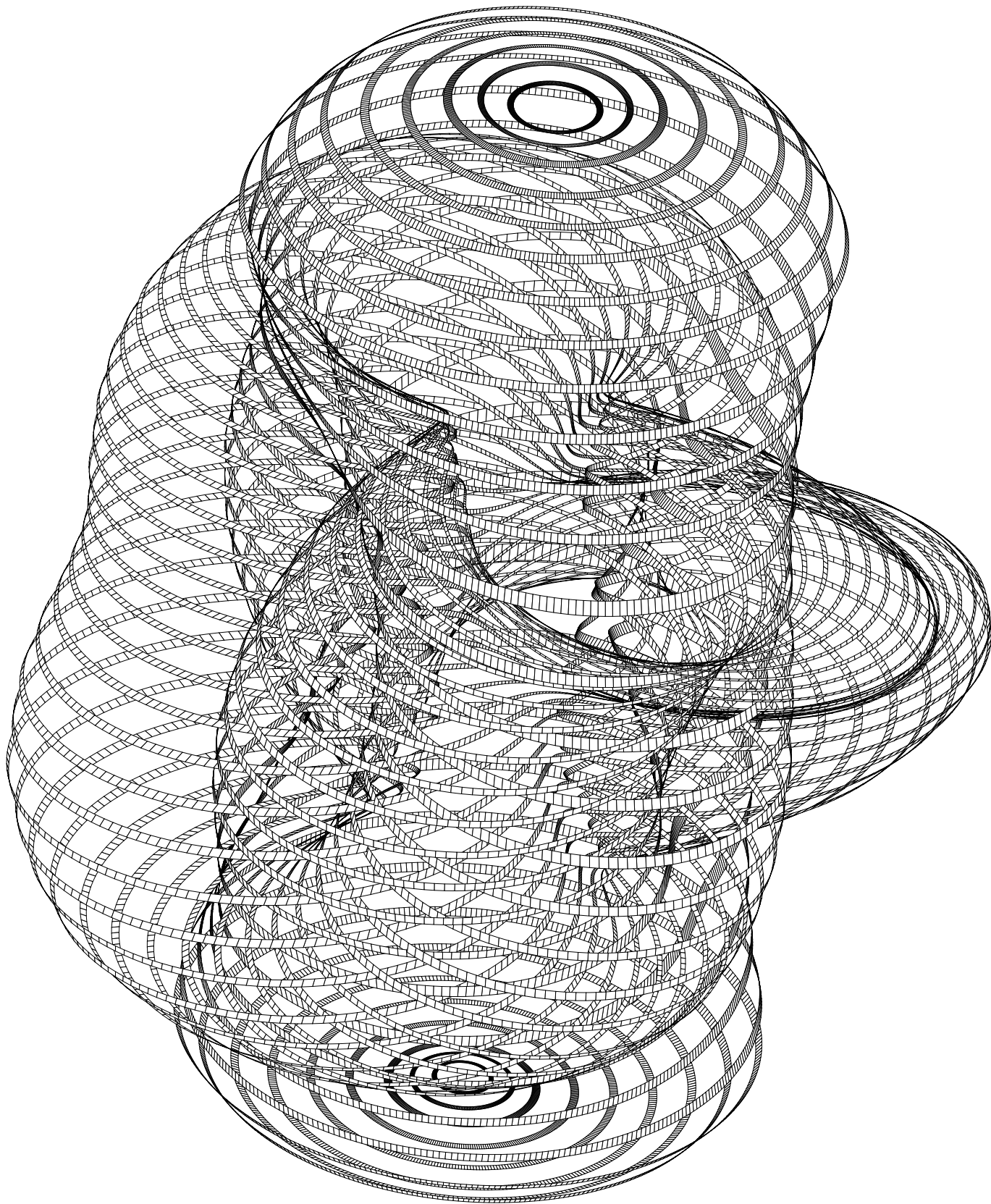
Figure 4.11: Projection of a +2-twist spun trefoil knot. The surface has transparent bands along lines of constant latitude.

through the sphere, each slice will intersect the sphere in a union of circles (contour lines). In the higher dimensional case, we have the extra possibility that the circles may be knotted or linked. Figure 4.12 shows an example of a movie for a spun trefoil. Unfortunately, the rendering on paper makes it difficult to clearly see what is happening. The parameter t can be thought of as "time"; it simply specifies the location of the 3D slicing hyperplane. From $t = -7$ to $t = 4$ the slice consists of two unlinked circles, they approach each other closely, and at $t = -3$ a "fusion" takes place. At this stage, the slice is a knotted graph in $\mathcal{R}^3$. Proceeding to $t = -2$, a "fission" occurs and the slice becomes a Square knot in $\mathcal{R}^3$. From this point on until $t = 0$ the slice remains a Square knot, and after $t = 0$ the entire sequence of slices occurs in reverse order.

Figures 4.13 and 4.14 show similar movies for the +1-twist and +2-twist cases, respectively. Here the deformations are somewhat more interesting. Also, the slice at $t = 0$ in Figure 4.13 is a Square knot again, so we can see that a cross-section of an *unknotted sphere* in $\mathcal{R}^4$ can be itself knotted (recall that all +1-twist spun knots are unknotted).

This method can also be used to construct arbitrary knotted 2-spheres in 4D including those which are neither twist-spun nor suspended. Some examples of these are shown in [Fox62, FR86, Lom83]. A future extension of KnotPlot will allow such constructions directly from a sequence of knots in $\mathcal{R}^3$.

$t = -7$ $t = -6$ $t = -5$

$t = -4$ $t = -3$ $t = -2$

$t = -1$ $t = 0$ $t = 1$

$t = 2$ $t = 3$ $t = 4$

$t = 5$ $t = 6$ $t = 7$
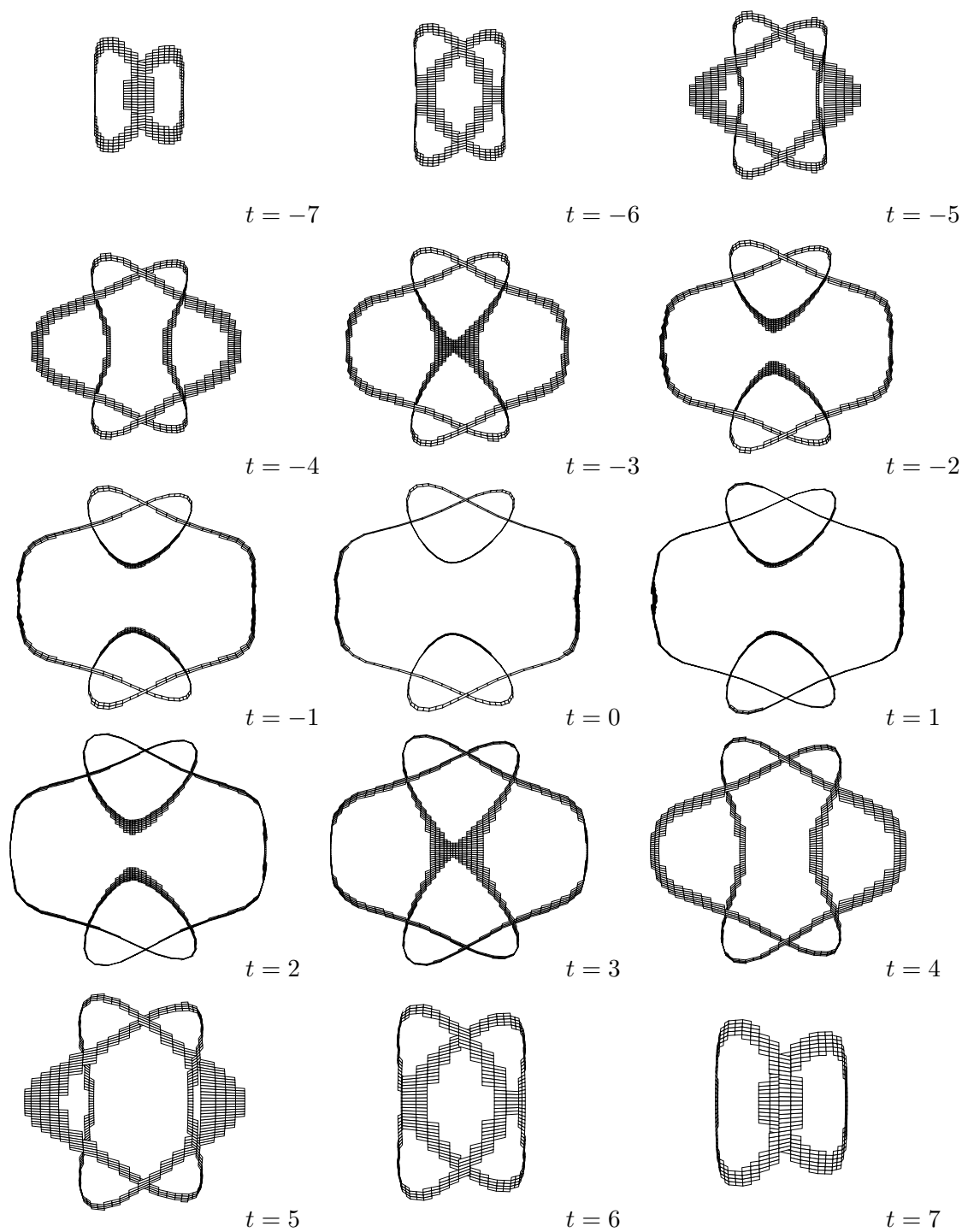
Figure 4.12: Movie of a 0-twist spun knot.

$t = -7$ $t = -6$ $t = -5$

$t = -4$ $t = -3$ $t = -2$

$t = -1$ $t = 0$ $t = 1$

$t = 2$ $t = 3$ $t = 4$

$t = 5$ $t = 6$ $t = 7$
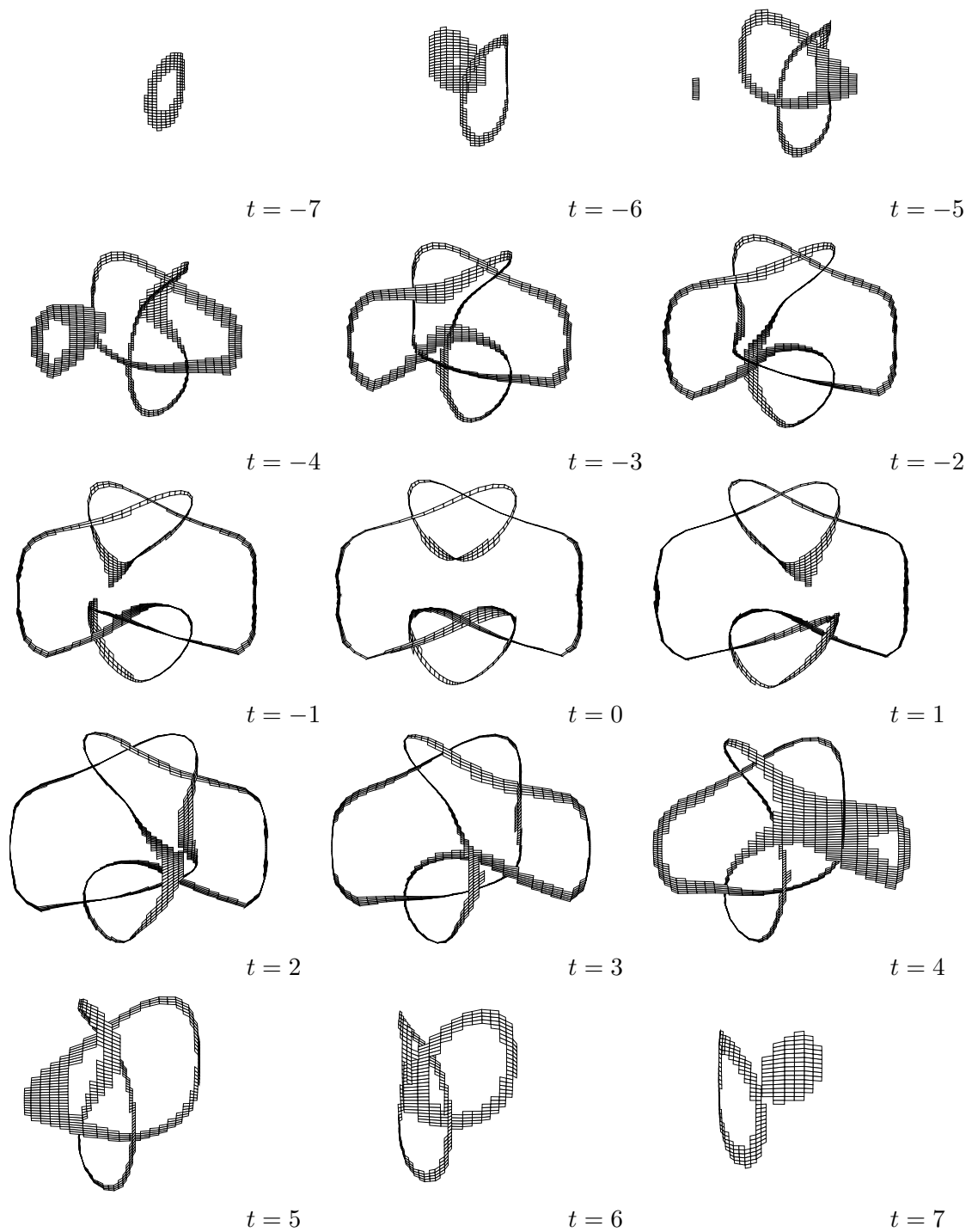
Figure 4.13: Movie of a +1-twist spun knot.

$t = -7$ $t = -6$ $t = -5$

$t = -4$ $t = -3$ $t = -2$

$t = -1$ $t = 0$ $t = 1$

$t = 2$ $t = 3$ $t = 4$

$t = 5$ $t = 6$ $t = 7$
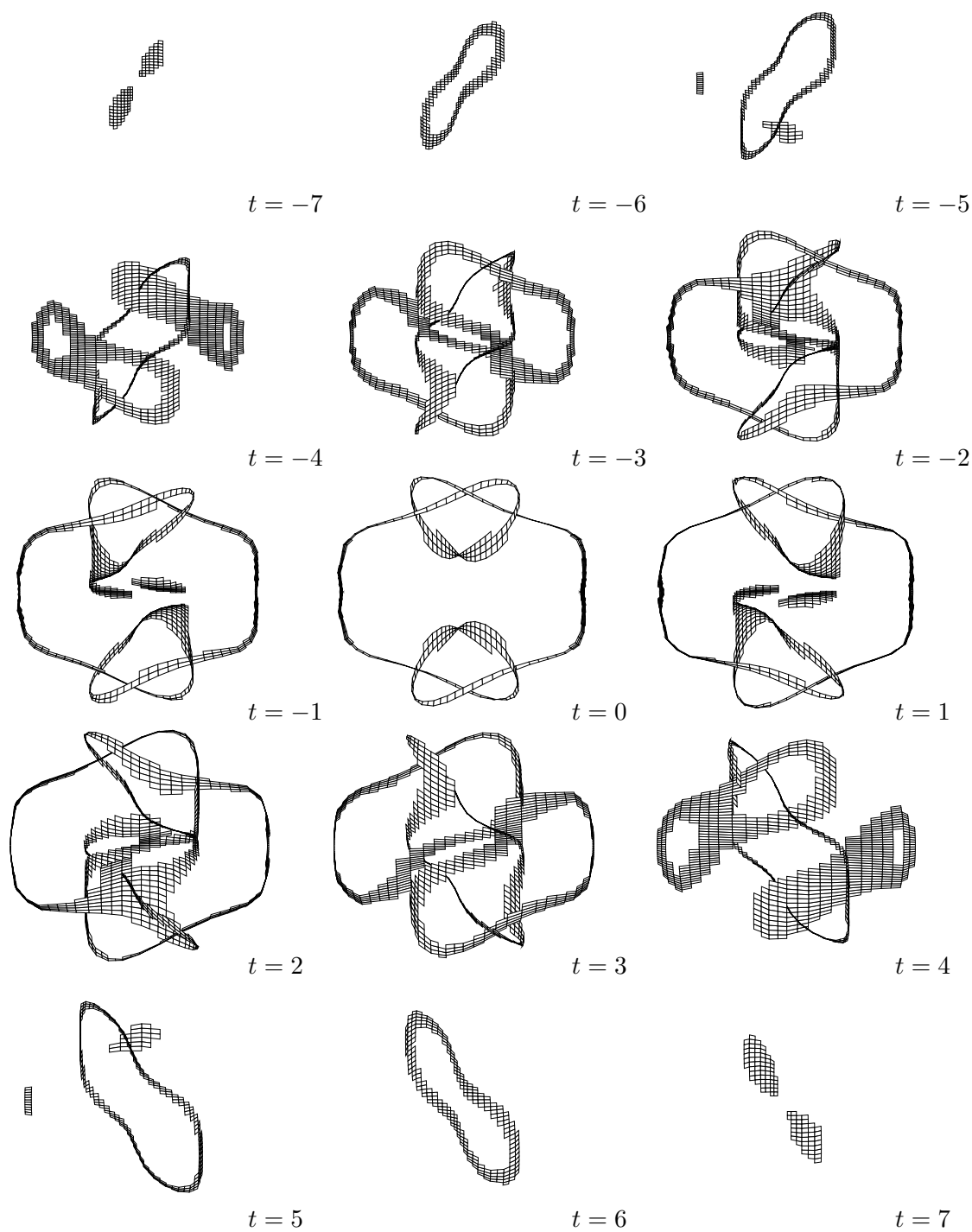
Figure 4.14: Movie of a $+2$-twist spun knot.

78

Chapter 5

Creating an initial embedding

This chapter is the first of two that describe methods for creating initial embeddings of knots and links. The methods described in this chapter are generally interactive, although they may rely on external software that performs its function without user interaction. In contrast, the technique described in Chapter 6 is almost entirely automatic and the user has little choice in the form of the actual embedding.

5.1   Knot catalogue

KnotPlot has available online over 1000 preconstructed knots[1] for use in visualization or experiments. About 80% of these are prime knots, the rest are "special" in some way, either by being interesting mathematically (for example common composite knots such as the square and granny knots), or by having a unique importance outside of knot theory. This latter group includes many decorative knots. Also in the catalogue, although created on demand, are several infinite classes of knots, notably the torus knots and knots obtained via closing a braid.

This large data set makes KnotPlot an interesting and useful tool for displaying many different knots even if no topological drawing is ever done. This could be useful for exploring knot theory, or for producing illustrations to be used in publications. Another use of the catalogue is

---

[1]We remind the reader of our convention of using the term "knot" to encompass both knots and links.

as a source of "raw material" for experiments using software other than KnotPlot. The papers by Katritch *et al.* [KBM$^+$96, KOP$^+$97] and Mao *et al.* [MSS97] largely fall into this camp, although each of these research groups also used KnotPlot as a construction and visualization tool in addition to using its data set.

5.1.1 Preconstructed knots

Prime knots

KnotPlot users have access to the entire catalogue shown in Appendix C of Rolfsen's book *Knots and Links* [Rol76]. The original data for these knots was entered into a computer by Margaret Flunkert and kindly provided by Prof. Rolfsen. The knots were relaxed according to the methods to be described in Chapter 7. Many knots occur in the catalogue in several forms.

good diagram   fully relaxed   minimal stick
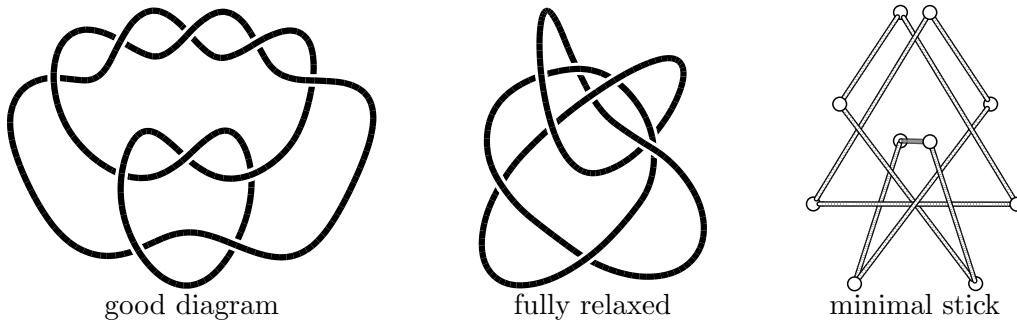
Figure 5.1: Three different views of the knot 10_{124}.

In Figure 5.1 we see the knot 10_{124} in three different forms. The first of these is an embedding that produces a "good diagram"; *i.e.* a minimal projection for the knot that contains only the allowed crossing types shown in Figure 3.2. The second version of 10_{124} shown is less satisfactory as a 2D projection but is a "better" 3D embedding, having been fully relaxed using the dynamics of Chapter 7. Typically, knots with higher crossing numbers (greater than six or seven) will not produce good 2D projections when fully relaxed. This observation is by no means rigourous or exhaustive and merits further study. The final version of 10_{124} is a minimal-stick candidate in a configuration that minimizes the minimum distance energy (see Section 3.5). This configuration has several curious symmetries that are discussed in more depth in Chapter 8.

On-going work is producing KnotPlot representations for the 281 10-crossing links with two or more components given by their Conway notation [Con70], although any one of these can be created on demand in KnotPlot by typing the Conway notation directly. As for the rest of the vast host of knots now catalogued by researchers [Thi97, Hos97], the user of KnotPlot must rely on the methods presented in Chapter 6 to construct an embedding for visualization purposes.

Special knots

The knot catalogue contains a large collection of knots which are not necessarily prime, but which have other qualities that set them apart. Some of these we have seen before, such as the square and granny knots of Chapter 3. Also included are knots that have a specific mathematical interest, such as the four-component Brunnian link[2] shown in Figure 5.2. Many of these knots

four component Brunnian link   Celtic knot   Ashley knot #2445
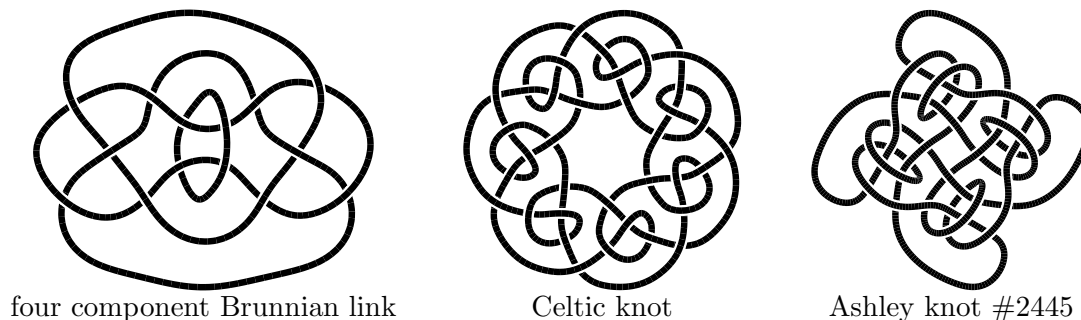
Figure 5.2: Some "special" knots from the catalogue.

are given as examples in *Knots and Links* [Rol76]. Other knots of mathematical interest in the catalogue are the 19 non-trivial knots of 16 or fewer crossings that have a trivial Alexander polynomial (constructed from data in [Thi85]), hyperbolic knots with "simple" complements (from the study of [CDW98], see Chapter 8), and numerous other examples culled from many papers in knot theory.

Other special knots in the catalogue are purely "decorative", such as the Celtic knot and the Ashley knot #2445 in Figure 5.2, taken from [Bai73] and [Ash44], respectively. Even

---

[2]A Brunnian link is a non-trivial link that becomes trivial with the removal of any one component. Another example is the Borromean rings shown on page 14.

though these knots are decorative, because of their symmetry they serve as useful examples of the construction techniques discussed later in this chapter.

5.1.2 Special classes of knots

Torus knots

Torus knots form an infinite family of inequivalent knots that in many ways are the "simplest" class of knots. For example, properties such as the unknotting number, crossing number and the braid index are known for all torus knots, whereas these properties are often difficult to compute for "typical" knots. Torus knots are those knots that exist as simple closed curves on the surface of a standardly embedded torus as shown in Figure 5.3. An (N, M)-torus knot,
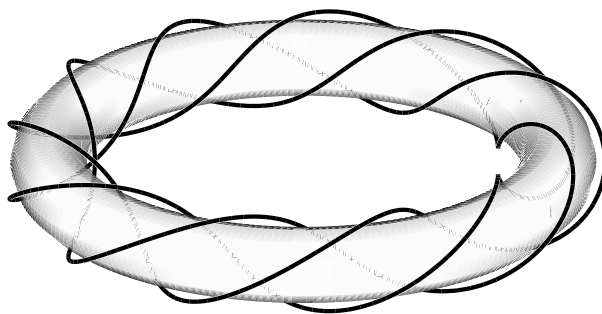
Figure 5.3: Torus knot $K_{3,11}$ as a simple closed curve on the surface of a torus.

denoted by $K_{N,M}$ is one that winds about the core of the torus N times while winding about the axis of the torus M times. The example shown in Figure 5.3 is for $N = 3$ and $M = 11$. If N and M are co-prime, non-zero integers, then a proper knot results. Unlike the vast majority of knots, the general torus knot $K_{N,M}$ has a simple parametric description

$$x = r \cos(N\theta), \quad y = r \sin(N\theta), \quad z = d \sin(M\theta), \quad r = R + d \cos(M\theta), \quad \text{for} \ \ 0 \leq \theta < 2\pi. \quad (5.1)$$

where R and d are the major and minor radii of the torus. If N and M are not co-prime, the convention is that $K_{N,M}$ represents the p-component link where p is the greatest common divisor of N and M. The torus link consists of p identical copies of the torus knot $K_{N/p,M/p}$, each offset by an angle $2\pi/p$ from one another.

One initially surprising fact is that $K_{N,M}$ and $K_{M,N}$ are equivalent knots. Upon reflecting on Figure 5.3 and the method of construction, this should become obvious.[3] Torus knots have many other surprising properties, for example the crossing number of $K_{N,M}$ is known to be $\min((N(M-1), M(N-1))$, and the stick number of $K_{M-1,M}$ is $2M$ [ABGW97]. The crossing number result allows us to generate a "zoo" of torus knots in order of (non-decreasing) crossing number, as shown in Colour Plate 1. Several familiar knots are torus knots, namely the Hopf link and the trefoil (the unknots $K_{1,1}$ and $K_{1,2}$ are not drawn). An excellent and rigourous discussion of torus knots appears in Murasugi's book *Knot Theory and Its Applications* [Mur96]. Therein the reader will discover that torus knots are the exception rather than the rule, and in a certain sense the torus knots have been completely classified. However, as the crossing number increases, the torus knots constitute an asymptotically zero fraction of all knots. Hence essentially all knots remain "difficult" to classify.

Knots from closed braids

Section 3.4 showed several examples of braid diagrams. General braids are easily created with KnotPlot by simply typing in the braid word. Since all knots can be represented as the closure of some braid [Ale23], in principle this method could be used to generate any knot. The difficulty is that it is usually not easy to determine the braid word corresponding to a given knot. Also, the knots obtained from closing a braid are often not in a minimal projection.

Random knots

KnotPlot has several methods for creating random knots. These are interesting in theoretical studies of the random knotting of long polymer chains of molecules [DPS, Pip89]. Figure 5.4 shows an example of a self-avoiding walk on a cubic lattice. Joining the endpoints results in a trefoil knot.

---

[3]Think of "unwrapping" the torus to form a rectangle in the plane with opposite edges identified, and consider the manner of curves drawn on the rectangle representing $K_{N,M}$ and $K_{M,N}$.
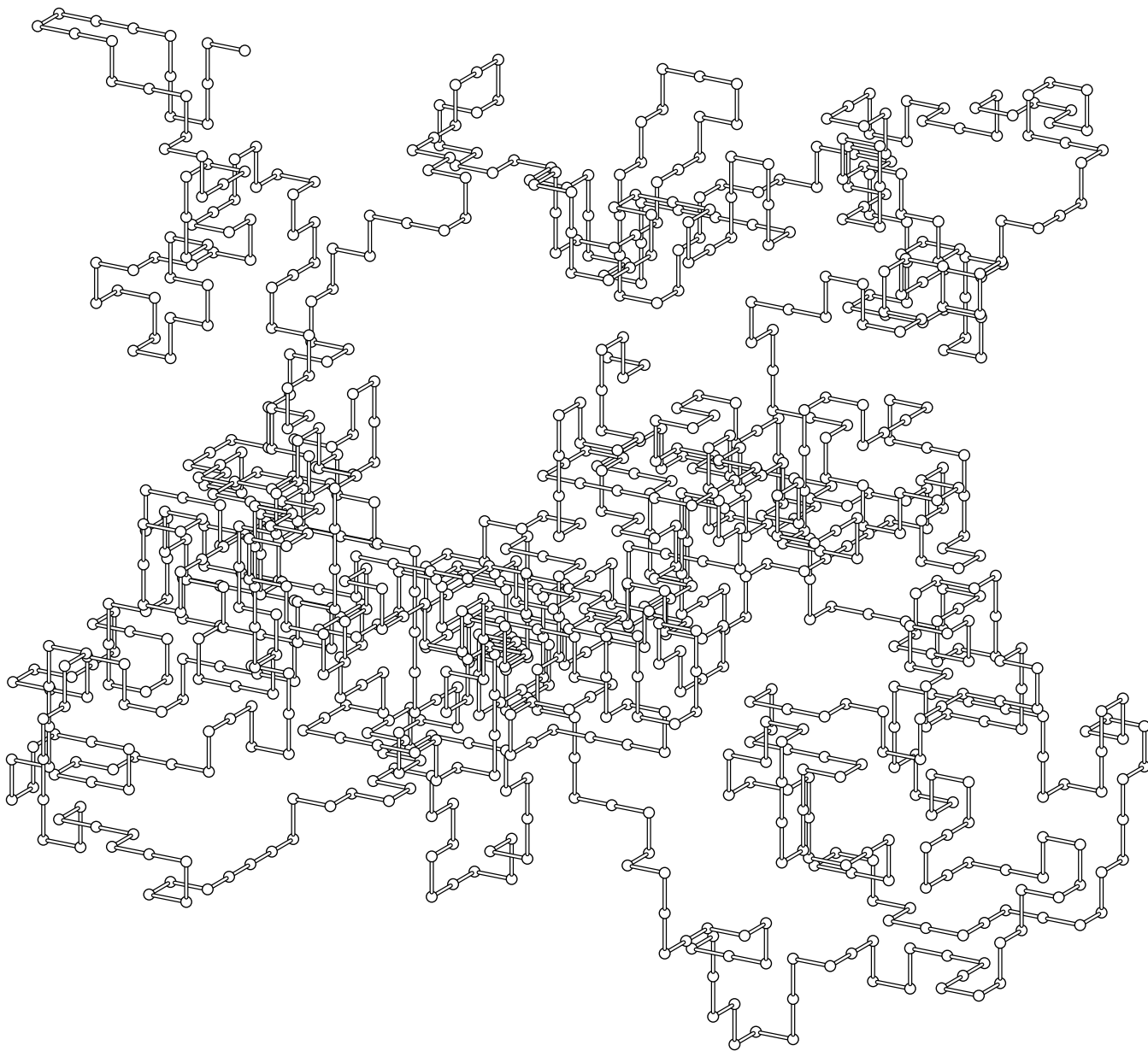
Figure 5.4: Self-avoiding random walk on the cubic lattice (1022 steps). Joining the endpoints in a straight line results in a left-handed trefoil.

Simple knot primitives: chains, lines, and circles

KnotPlot has several methods to create simple knot types, such as knot chains with an arbitrary
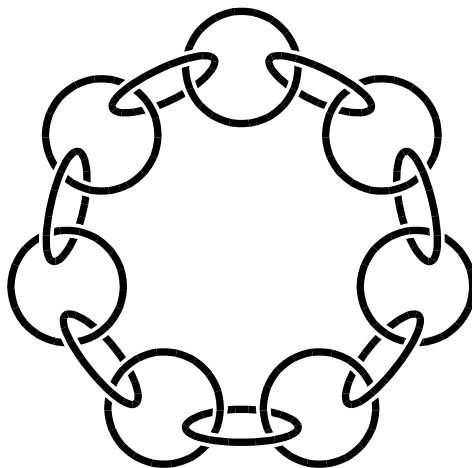
Figure 5.5: Linked chain of 14 unknots.

number of components as shown in Figure 5.5 or (even simpler) knot primitives such as (open-ended) lines and unknots (geometric circles) of various sizes and orientations. With the exception of the knot chains, these primitives are not of interest in themselves, but rather they are building blocks for more complicated objects. A specific example of complicated knots being assembled from a collection of primitives is given in the section on Lorenz knots later in this chapter.

5.2   Tangle calculator

KnotPlot's tangle calculator is a stack-based calculator for tangles. Elements on the stack are four-input tangles of arbitrary complexity. The stack may contain any number of tangles A, B, C .... The stack is imagined "growing upward" with the most recently created tangle, A, being on the "bottom" of the stack. The tangle calculator does not use the Conway notation directly. The conversion from the Conway notation to tangle calculator commands is a simple parsing problem the details of which are discussed in the KnotPlot manual [Sch97b]. Tangle calculator commands can be (and often are) entered directly without using the Conway notation. This is
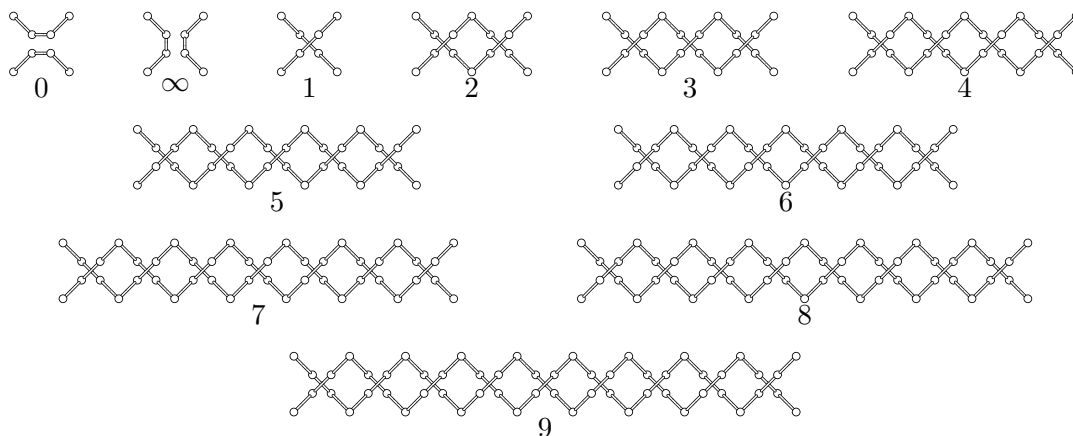
Figure 5.6: Tangle calculator primitives.

useful for users who are not sure of the interpretation of a given string in the Conway notation (which is a little tricky in certain ways), but who find the tangle calculator intuitive and simple. Also, it allows for the creation of knots which do not have any corresponding Conway notation.

The tangle calculator has two basic modes of operation. The first is a direct mode where the user types in a "tangle string", which is a string of tangle commands that are all executed in sequence without any possibility for user intervention. The second mode is an interactive mode. In this mode, the computer screen displays the bottom few tangles on the stack as well as a "keypad" with tangle calculator commands that the user can choose by clicking the mouse.

5.2.1 Basic operators

Primitives

The tangle calculator behaves much like a Reverse Polish Notation (RPN) calculator, except that the operands are four input tangles. Like any calculator, there are operators for creating primitive objects. For the tangle calculator, these are the fundamental and integral tangles shown in Figure 5.6. The tangle calculator commands corresponding to each of the primitives shown in Figure 5.6 will be denoted by 0, 1, ... 9 except for the ∞-tangle which will be denoted by i. The effect of each of these commands is to push the tangle stack upwards and insert the appropriate tangle as the bottom element A of the stack.

86

Binary operators

Only two binary operators are defined in the tangle calculator. These correspond to the direct sum and product operators shown in Figure 5.7 (seen before as Figure 3.9). Each of these binary

+ operator (direct sum)   * operator (product)
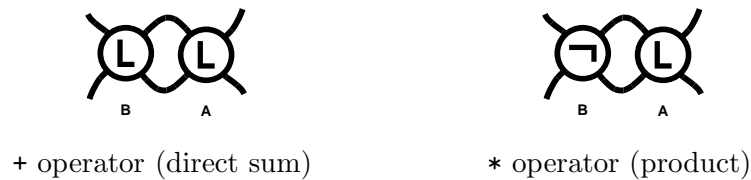
Figure 5.7: Tangle calculator binary operators.

commands has the effect of replacing the bottom two elements of the tangle stack A and B with the result of the operation that becomes the new bottom element A of the stack.

Unary operators

The tangle calculator's unary operators modify the bottom element of the stack A, leaving the rest of the stack unaffected. Most of these operators are simple: x, y, and z reflect A in one of the three coordinate axes, and the @ command rotates A by a quarter turn about its centre. Only slightly more involved are the commands + and -, which perform respectively the twist

original tangle   x   y   z   r   @   +   -
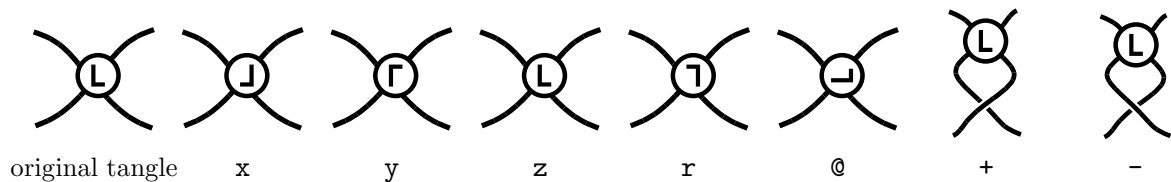
Figure 5.8: Tangle calculator unary operators.

operations shown in Figure 3.9, and the r operator which reflects A about a diagonal. These operators are shown schematically in Figure 5.8.

Stack operators

Finally, the tangle calculator provides the usual set of stack-based operators. These are the operators for "rolling" the stack up (<) or down (>), duplicating (!) or deleting (˜) the bottom

element of the stack, and exchanging (`e`) the bottom two elements of the stack.

5.2.2 Tangle templates

The previous section described the basic commands used in creating tangles. After a sequence of these commands has been executed, the tangle calculator will generally have one or more tangles on the tangle stack that may be used to create knots. For this we must embed the tangles into one of the *basic polyhedra* described in Section 3.2. Of the nine basic polyhedra described by Conway [Con70], only the $1^\star$ polyhedron and a variation of the $6^{\star\star}$ polyhedron are built into KnotPlot. The $1^\star$ polyhedron is straight-forward; Figure 5.9 shows how a 3-tangle is embedded

Trefoil   Borromean rings
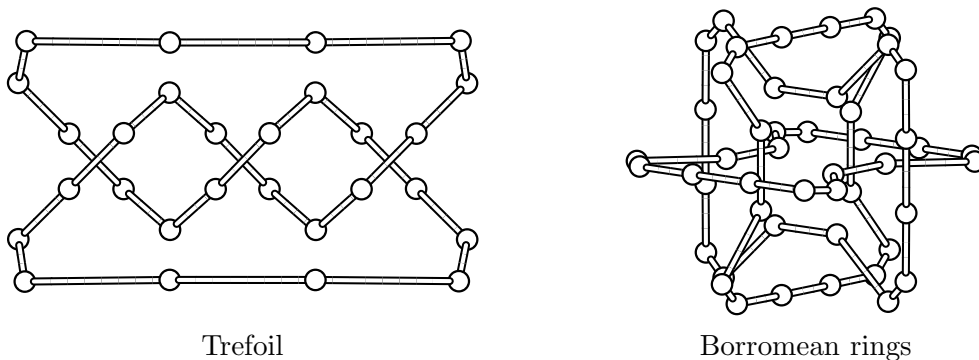
Figure 5.9: Some tangle calculator creations.

into the $1^\star$ polyhedron to form a trefoil knot. The variation of Conway's $6^{\star\star}$ polyhedron reveals the symmetry of that polyhedron by positioning the place-holders into which the tangles are inserted onto the faces of a regular cube in 3D rather than the flat 2D representation given by Conway. Figure 5.9 shows that the Borromean rings result if a 1-tangle is inserted into each of the tangle "slots". Although this variation is topologically equivalent to Conway's $6^{\star\star}$ polyhedron, it does not readily produce the same knot diagrams.

Given that the variation on the $6^{\star\star}$ polyhedron is (geometrically) different from the original one given by Conway, it is correct to state that KnotPlot cannot automatically create any knot from the Conway notation that uses a polyhedron other than the $1^\star$ polyhedron (*i.e.* if one expects to arrive at a minimal crossing diagram for the knot). This is not a limitation since the remaining polyhedra, and many more besides, may be interactively sketched, refined,

88

and manipulated in 3D by the *user*. The user does this by creating a *tangle template*. A tangle template has the same function as one of the basic polyhedra, however, the term has a specific technical meaning determined by the manner in which such templates are implemented
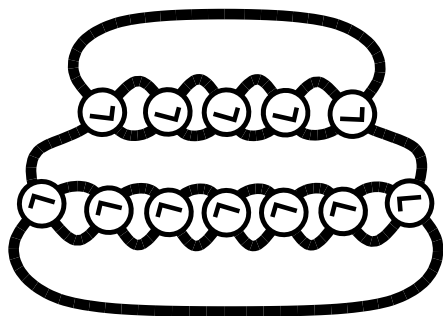
Figure 5.10: Example of a tangle template.
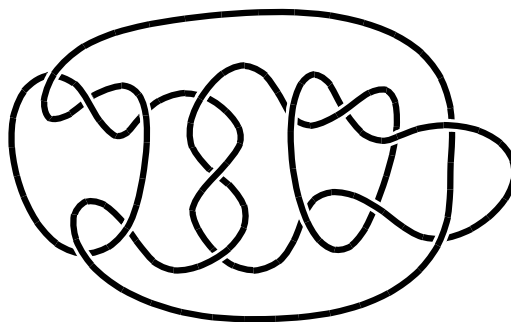
Figure 5.11: 15 crossing mutant of a 13 crossing knot created with the tangle calculator.

in KnotPlot (also the term is more intuitive than the term basic polyhedra). Tangle templates are special kinds of knots represented in the usual manner in KnotPlot, augmented with one or more *tangle holders*. Tangle holders are four beads in the knot that are geometrically close to each other and each located at the endpoint of an open knot. An example of a tangle template is shown in Figure 5.10. KnotPlot has commands for creating a tangle template from a knot. The details of this are described in the KnotPlot manual [Sch97b].

5.2.3   Example: creating a "mutant"

To close this section on the tangle calculator, we will illustrate the creation of a non-trivial knot, namely the 15-crossing "mutant" of a 13-crossing knot shown in Figure 5.11 (from [Thi85]). Figure 5.12 shows the steps taken in creating the mutant (some of the operations have been coalesced) together with the state of the tangle stack. The final step embeds the tangle into the $1^\star$ polyhedron. After a small amount of refinement we arrive at the smoother embedding shown in Figure 5.11.
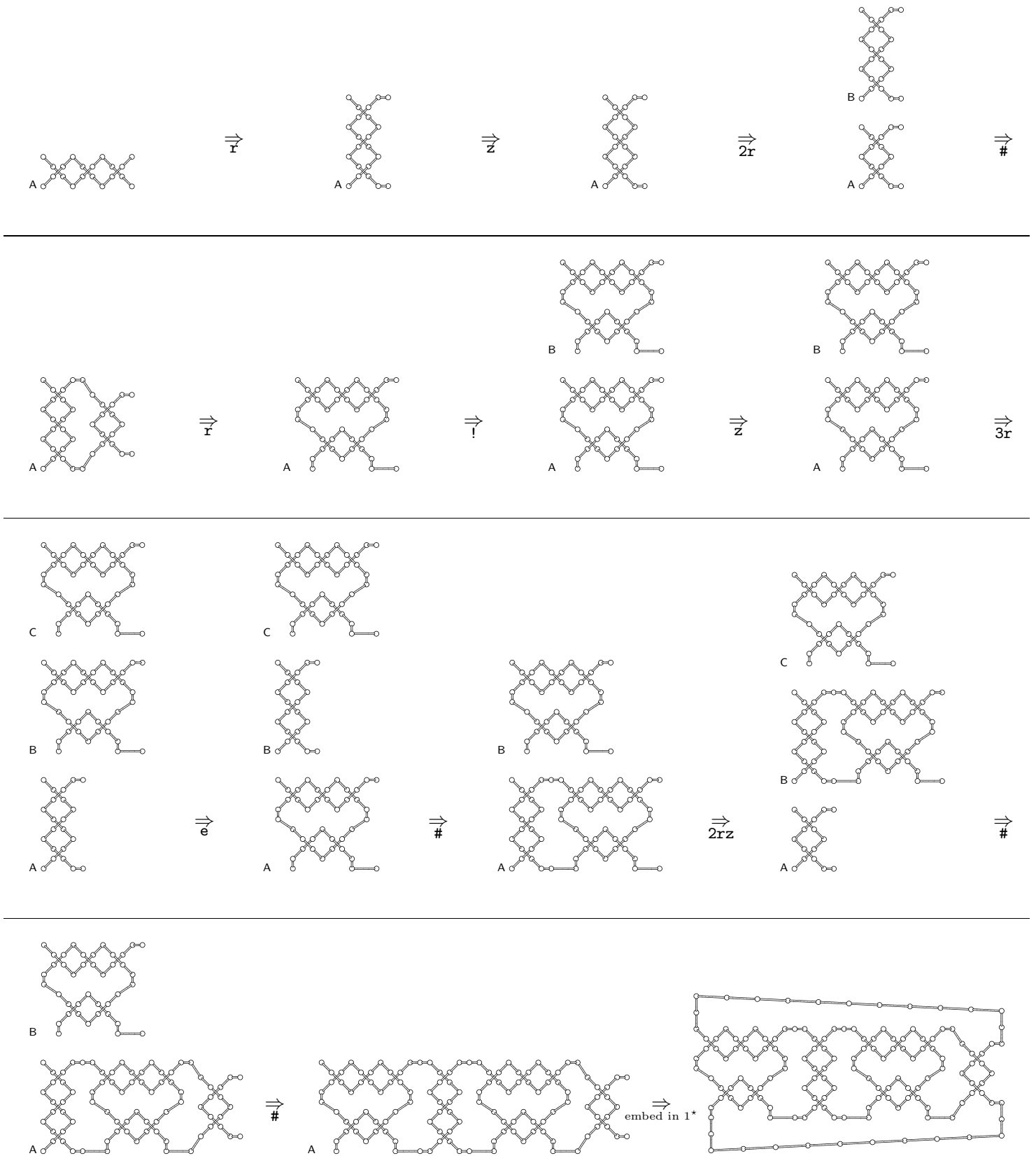
Figure 5.12: Creating the mutant with the tangle calculator.

5.3   Sketching

Even though the previous methods allow for the creation of many knots, it is often desirable to create knots with arbitrary embeddings. For this purpose, sketching is essentially the only alternative. KnotPlot employs a simple sketching technique. Depending on the patience of the one doing the sketching, this technique may not result in "acceptable" results. Generally such sketches are refined using the methods described in Chapter 7. Often only a small amount of refinement can greatly improve the embedding.

Sketching a knot with KnotPlot involves merely clicking down with the mouse at the desired bead locations. The sketching is done in a plane, with the exception that over- and under-crossings may be indicated (by using different mouse buttons). For this reason, this form of sketching is often described as "$2\frac{1}{2}$D" rather than 3D. Actual 3D sketching is rarely needed. In any event, the 3D placement of the knot may be altered with the topological refinement methods of Chapter 7.

One useful technique is to sketch with an image of a knot displayed as a backdrop. The image would usually be derived from a scanned photograph, or a hand drawing of a knot. It could also be a crude picture of a knot made with a paint program. Figure 5.13 shows a (simulated) view that a user would see while sketching a knot with a backdrop (backdrop shows knot #2445 digitally scanned from [Ash44]).

5.4   Construction and deformation tools

KnotPlot has a wide variety of construction tools. These include methods for translation, scaling, and rotation as would be expected for any program that can edit objects in 3D. Specific to knots are additional methods for cutting and splicing, transforming individual components, and the direct manipulation of the knot's embedding using the mouse. Essentially all of these methods are trivial and will not be discussed here. Again, the reader is advised to refer to the KnotPlot manual [Sch97b] for more complete information.

The remainder of this section will discuss in detail two of the mathematically more
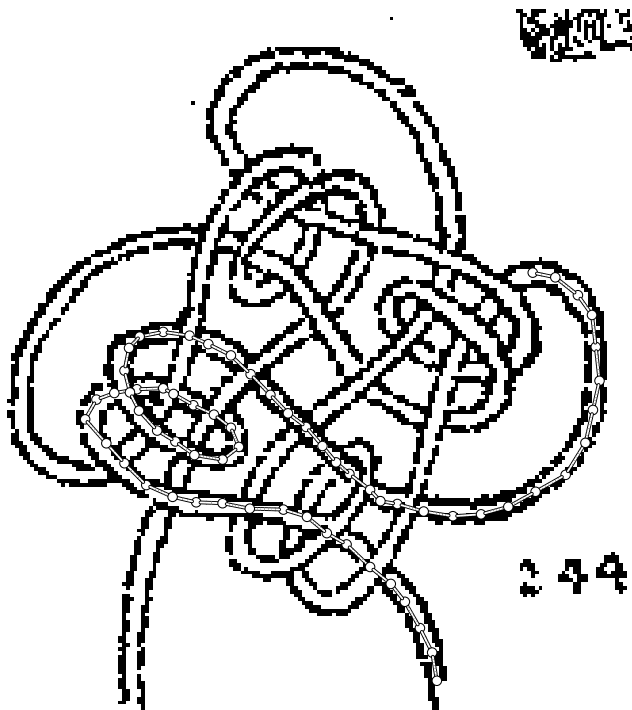
Figure 5.13: Sketching a knot with a backdrop image.

interesting construction techniques, symmetry operations and warping.

5.4.1 Symmetry operations

Figure 5.2 shows several decorative knots constructed using a symmetry operation. Both of these knots can be assembled from smaller pieces which are then assembled under one of the point groups.

5.4.2 Warping

KnotPlot's warping mechanism is similar to what is called "free-form deformation" in the computer graphics literature [Bar94, Coq90]. In typical applications of free-form deformation, the issues are somewhat different than we are concerned with, as the objects deformed are generally surfaces or lattices of points in $\mathcal{R}^3$. An example of a topic that is of concern in the deformation of lattices is volume conservation, which is irrelevant to topological drawing. For our purposes,

warping will simply mean applying a mapping (not necessarily a homeomorphism) from $\mathcal{R}^3$ into $\mathcal{R}^3$. It is usually the intent that these mappings change the knot type.

Rectangle warping

The simplest form of warping maps the rectangle of height h and width w centered at the origin into an annulus as shown in Figure 5.14. This transformation affects only the x and y coordinate values, the z value is unchanged. As a mapping $(x, y, z) \mapsto (x', y', z')$, the warp can be written
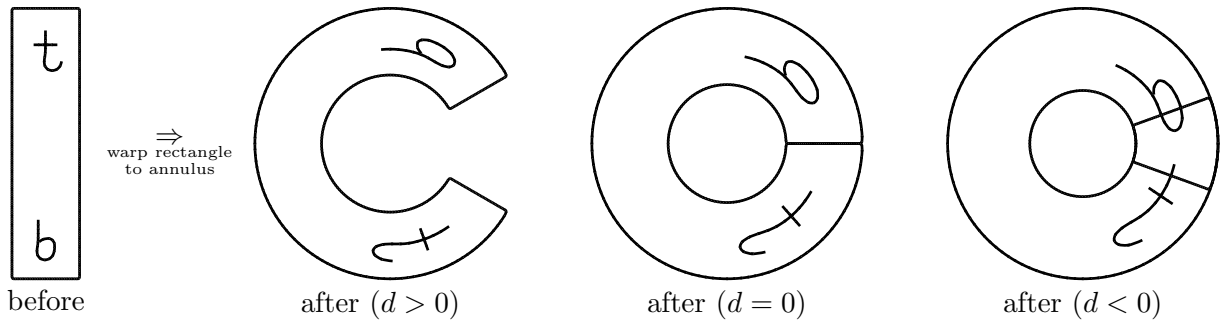
Figure 5.14: Rectangle warping.

as
$$x' = r \cos \theta, \quad y' = r \sin \theta, \quad z' = z, \quad \text{where} \quad r = x + \frac{h+d}{\pi}, \quad \theta = \frac{2\pi}{h+d}(y - \frac{h+d}{\pi}).$$

The parameter d controls the amount of overlap after the warp; there is no overlap if $d > 0$. If $d = 0$, then the top and bottom edges of the rectangle exactly coincide after the warp. A typical application with $d > 0$ is to create a circular closed braid as shown in Figure 5.15. The
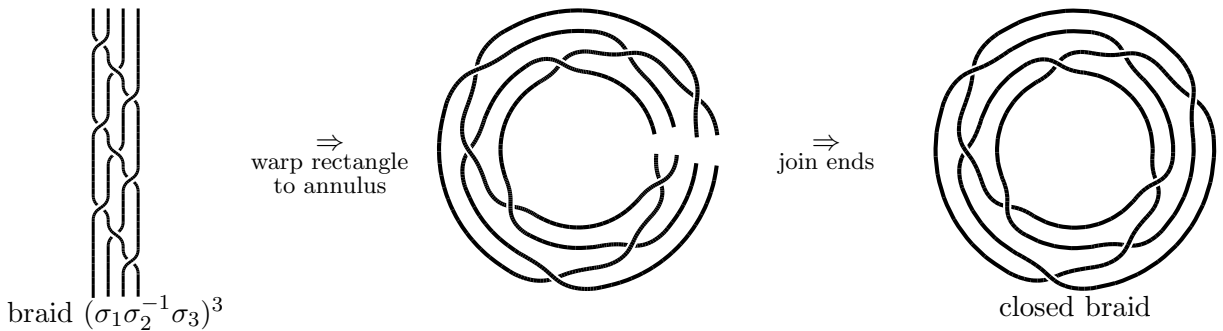
Figure 5.15: Creating a closed braid with rectangle warping.

93

x-branch before warp

y-branch before warp

after warping   after join
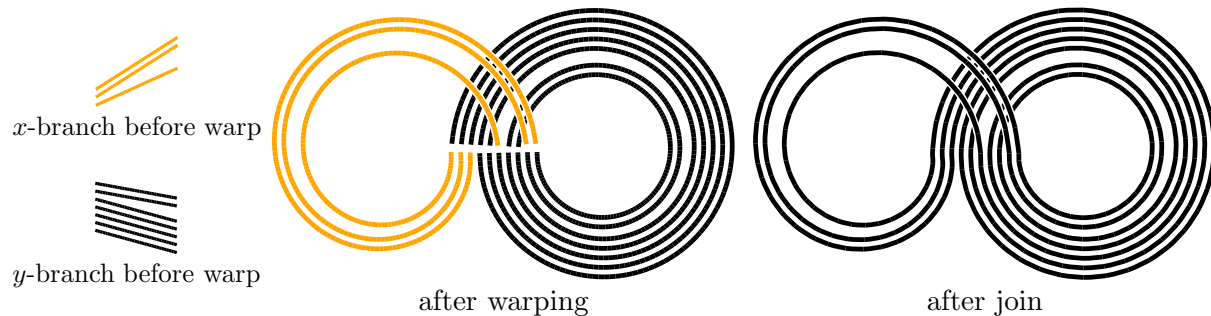
Figure 5.16: Lorenz warping to create knot with word $xy(xy^3)^2$.

remaining case $d < 0$ will cause the rectangle to self-intersect after being warped. An application is shown in Figure 5.19 where a value of $d < 0$ is used to create a twist knot.

Lorenz warping

Similar to rectangle warping is Lorenz warping. This specialized form of warping is only useful for creating Lorenz knots (Section 3.5.3). Two forms of the Lorenz warp map a square centered at the origin into the x and y-branches of the Lorenz template as shown in Figure 5.16. The objects being warped are collections of lines as shown on the left of Figure 5.16, created with the line primitive previously mentioned. Each line corresponds to one symbol in the Lorenz word denoting the knot. A simple algorithm described in [Hol88] is used to generate the correct placement of these lines. This example is a good illustration of the utility of simple primitives in generating more complex objects. Figure 5.20 at the end of this chapter shows a rather complicated Lorenz link of five components created with this method (one component is the same as the knot in Figure 5.16).

Warping along a link

This form of *warping along a link* is much more powerful than the previous forms discussed; it includes both as special cases. It does, however, require more setup than the other methods and so is used only when the other methods are not sufficient. The idea here is to use a two component link as an auxiliary construction device. The most useful for this purpose are the

"doubled knots" shown in Figure 5.17. Each of these links has one component (drawn in black)

trefoil ($w = 3$) | knot 6_3 ($w = 0$) | knot 9_{42} ($w = 1$) | knot 10_{134} ($w = 4$)
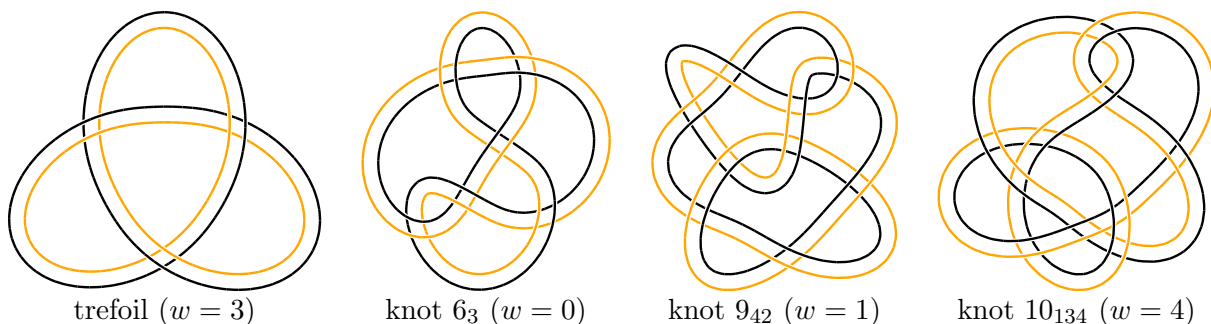
Figure 5.17: Doubled knots in a blackboard framing.

together with a second component of the same knot type (drawn in a lighter colour) that lies close to the first component. If we imagine filling the space between these two components, we arrive at an embedded ribbon in space of the same knot type as one of the components. The boundary of this ribbon is the two components themselves. If this ribbon can be "pressed flat" onto a plane (except for the necessary crossings), then it is said to be in a *blackboard framing* [Kau91]. More formally, if the two components have the same orientation, then the blackboard framing is obtained when the linking number of the components is equal to the writhe of a single component.

We may then use the ribbon constructed from one of these doubled knots as a deformation path. To warp along the link, we map a circular annulus in the xy-plane onto the ribbon itself. The original z-component is used to specify an offset from the ribbon in the direction of the local normal of the surface. If we start with a torus knot and warp along a blackboard framed knot, then we can create all of the *cabled knots*, examples of which are shown in Figure 5.18. In these examples, the four component torus link $K_{4,44}$ is the initial object created.

Cabling is an example of a general procedure in knot theory of creating *satellite knots*, introduced by Schubert [Sch53]. In the cabled trefoil of Figure 5.18, the torus link $K_{4,44}$ is called the *satellite* of the trefoil knot, which is its *companion*. Chapter 8 will show another satellite construction using several features of KnotPlot.

With this form of warping, it is not necessary to start with a link of the form shown

trefoil 7_6
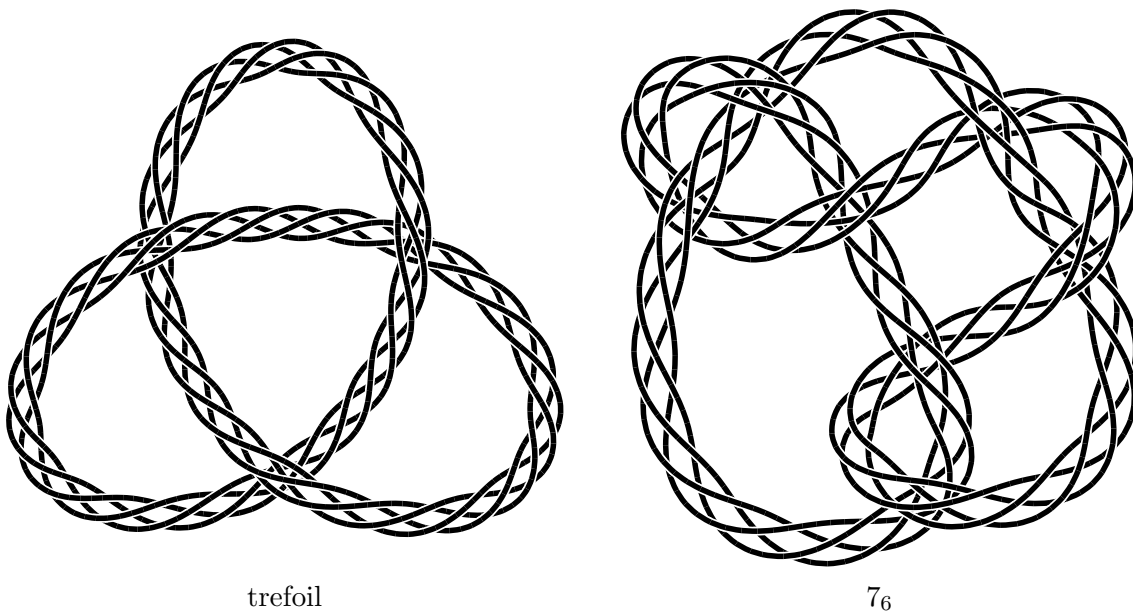
Figure 5.18: Examples of cabled knots with cabling $(4, 44)$.

in Figure 5.17. Arbitrary two-component links may be used to define the warp path. In this case, the results may be unpredictable. The Lorenz warp operation could be implemented as a combination of rectangle warping (to warp the initial squares onto an annulus) followed by a warping along a link whose components lie along the boundary of the Lorenz template.
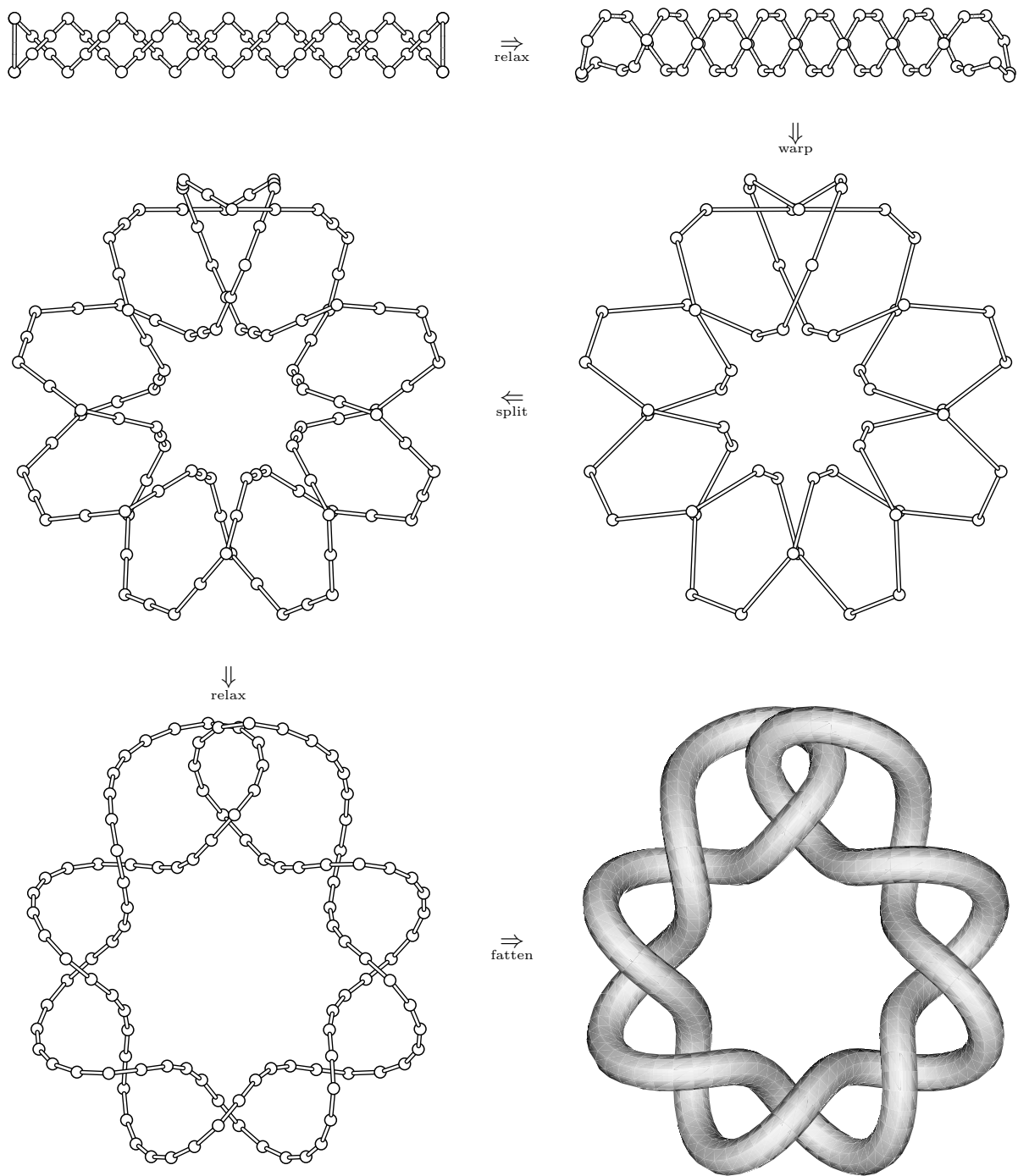
Figure 5.19: Creation of the twist knot 9_2 using the tangle calculator and warping operation.
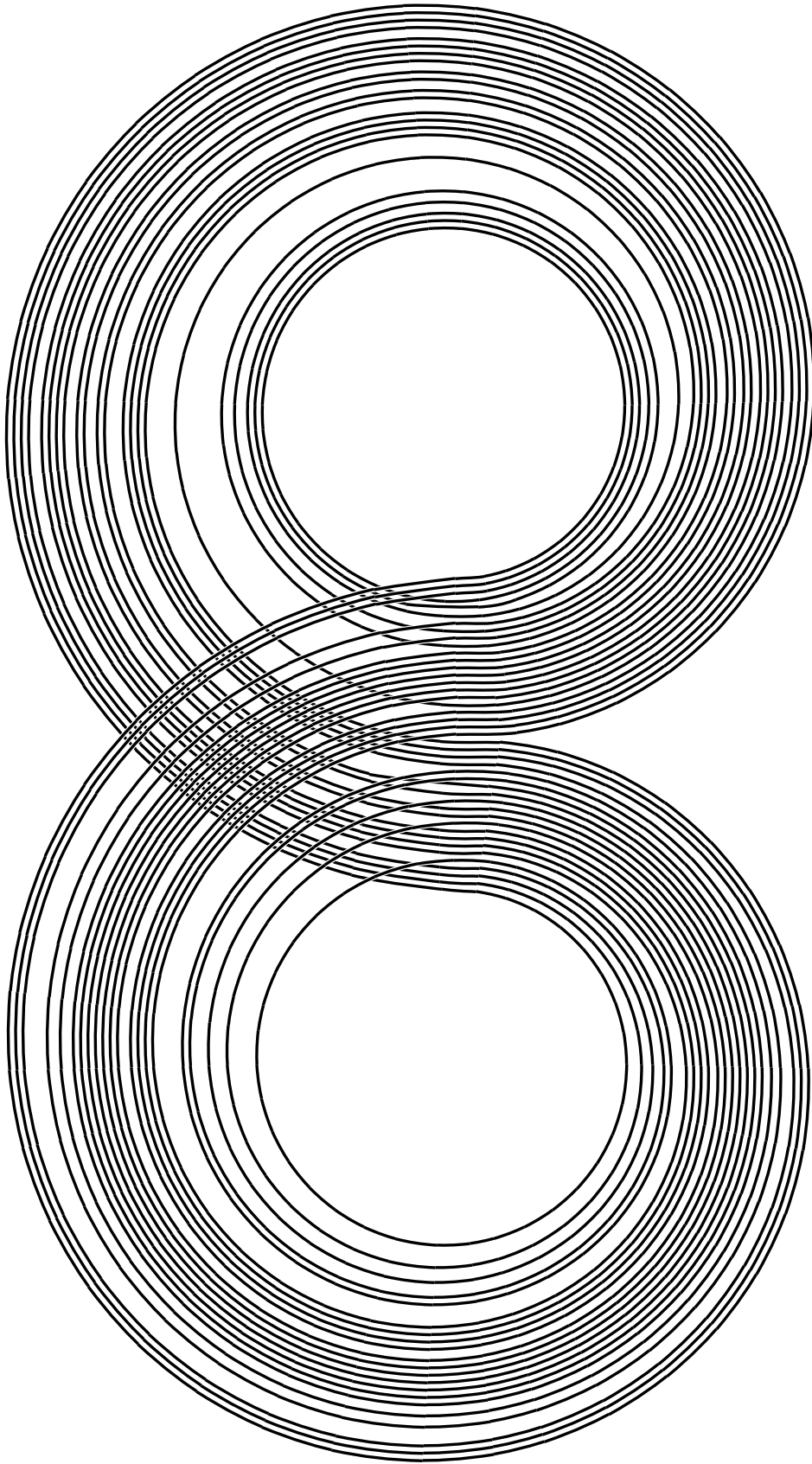
Figure 5.20: Link of Lorenz knots with words $x(yx)^2$, $x(yx)^3$, $xy(xy^3)^2$, x^3yxy^3xy and $x(yx)^5$.

Chapter 6

Dowker-Thistlethwaite codes

This chapter focuses on one method of providing an initial embedding of a knot. Such an embedding may be refined using techniques described in Chapter 7. Here we are only concerned with constructing the embedding from what is purely a topological description of the knot. To this end, we will use graphs as an auxiliary device for working with knots. Since knot diagrams in the plane are essentially 4-valent planar graphs with labelled vertices indicating the crossing type, it is interesting to consider what might be gained by viewing knots as graphs in two dimensions, rather than as three dimensional objects. Two applications of this approach are described below.

The first application is a useful method for obtaining knot diagrams from the tabulation of proper knots with 13 or fewer crossings by Thistlethwaite [Thi85], recently extended to 16 crossings by Hoste, Thistlethwaite, and Weeks [HTW98]. These knots are known only by an encoding that fully describes the knot as a topological object, but that doesn't directly indicate anything about its geometric placement. Knot diagrams have not been drawn for the

Crossing number	3	4	5	6	7	8	9	10	11	12	13
Number of knots	1	1	2	3	7	21	49	165	552	2176	9988

Table 6.1: Number of prime knots up to 13 crossings.

vast majority of the knots with more than 10 crossings. As can be seen by referring to Table 6.1, the number of prime knots with a given number of crossings grows rapidly with the crossing number, so there are many knots that have never had the privilege of being drawn. In fact the growth is known to be exponential [ES87], as might be expected by considering that there are 2^n ways to label a given 4-valent graph with 2 types of labels (left or right handed crossing).

The second application moves from 2D into 3D in an attempt to improve the subjective quality of the embeddings. This method may be used as an alternative to the technique of Chapter 7, which involves relaxing knots in their usual form of a polygon embedded in three dimensions.

6.1 Knot diagrams from Dowker-Thistlethwaite codes

A *Gauss code* for a knot diagram in the plane is a string of letters that encode the crossings. For an n crossing knot the Gauss code has $2n$ letters, each letter occurring twice. To generate the code, first label all the crossing points in the plane, for example with the letters A, B, C, ... as shown in Figure 6.1 for three different knots. Next pick an orientation and a starting point on

ABCABC   ABCDBADC   ABCDEFGCDEHABGFH
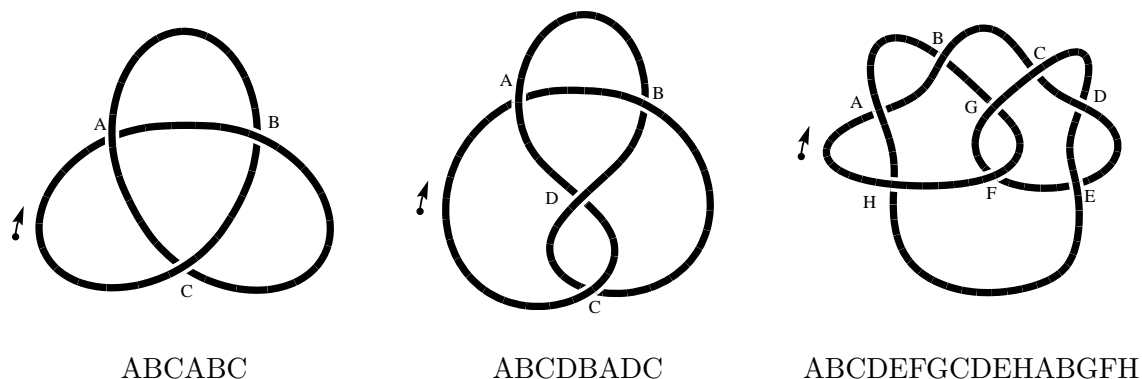
Figure 6.1: Several knots along with their Gauss codes.

the knot diagram and proceed around the knot while recording the label of each crossing as it is encountered. After returning to the starting point, a sequence of $2n$ labels, each one occurring twice, will be accumulated. This sequence completely encodes the knot diagram (up to mirror

images). The example for the trefoil gives the sequence ABCABC. During this procedure, no record is kept of whether the crossings are over or under-crossings. This limits the Gauss code as it stands to knot diagrams that are *alternating*; *i.e.* that alternate under, over, under... as one proceeds along the knot. However, upon reaching the 8-crossing knots, we start to encounter knots that are non-alternating in a minimal projection, so it is necessary to augment the Gauss code somewhat.

A simple variation of the Gauss code that can accommodate the non-alternating case was developed by Hugh Dowker and has proved invaluable in Dowker and Thistlethwaite's enumeration of knots with up to 13 crossings [DT83, Thi85]. The *Dowker-Thistlethwaite code* (or DT code) for a knot projection is obtained in a similar way to the Gauss code. Again pick a starting point on the diagram and proceed in one direction around the knot, this time labelling each crossing in order with the labels $1, 2, 3, \ldots 2n$ for an n-crossing knot. Each crossing will now receive two distinct labels. It is not hard to see that one label will be even and one odd. This procedure defines a parity reversing mapping $p(i)$ on the integers $1, 2, 3, \ldots 2n$ into themselves such that crossings i and $p(i)$ project to the same point on the plane and $p(p(i)) = i$. Because each crossing receives one odd and one even label, the mapping $p(i)$ is completely specified by the sequence $p(1), p(3), \ldots p(2n-1)$. If we augment each term of this sequence with a + or − sign depending on whether or not crossing $p(i)$ is an over or under crossing we obtain the DT-code of the knot. Several examples are shown in Figure 6.2. Obviously, the code may depend upon

$[4\ 6\ 2]$ $[6\ 8\ 2\ 4]$ $[12\ 8\ 10\ -14\ 4\ 16\ 2\ -6]$
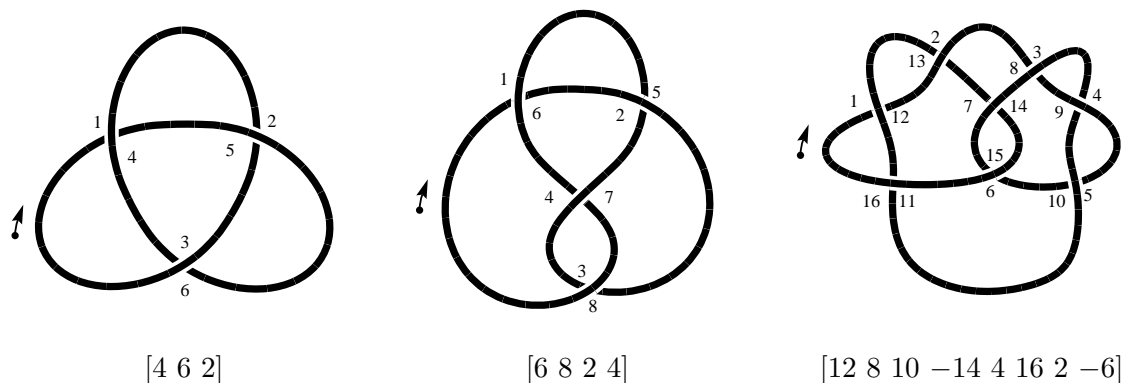
Figure 6.2: Several knots along with their Dowker-Thistlethwaite codes.

the starting point and the direction traversed (indicted by a dot and arrow in the figure). For example, reversing the orientation on the trefoil in Figure 6.2 gives the code $[-4 \ -6 \ -2]$, which in this case merely flips all the signs. The code doesn't depend on the starting point for this example because of the symmetry in the trefoil's diagram. However, in the case of the "true-love" knot on the right in Figure 6.2, all 16 possible choices for starting point and orientation give different codes. In addition this knot has more than one minimal crossing diagram (recall Figure 3.19, which shows two other minimal diagrams for this same knot), each of these yielding several more distinct codes. Although most knots (knot diagrams) have many DT-codes, one of these will be minimal in a lexicographical sense and that one may be chosen as the standard "name" for the knot (diagram). The enumeration in [DT83] follows this convention, however, none of the methods described in what follows are affected by which code is used.

Granny knot   Square knot in alternating projection
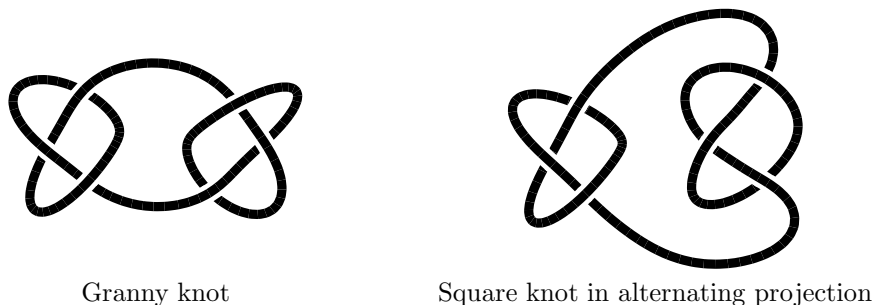
Figure 6.3: Different composite knots with the same DT-code of [10 6 8 4 12 2].

It is not immediately apparent that a DT-code uniquely specifies a given knot. We will see later the reason for this, at least for prime knots. As noted by Adams [Ada94], the DT-code does not uniquely specify *composite* knots, as shown in Figure 6.3. For more on this issue, as well as the problem of knowing when a given code actually specifies a knot, the reader is referred to the paper by Dowker and Thistlethwaite [DT83], where an algorithm is given to filter out invalid cases. If we assume that we are given a valid DT-code of a prime knot, the problem of generating a knot diagram from the code is readily accomplished by considering the 4-valent graph related to the knot diagram.

Several of these graphs are shown in Figure 6.4, for the same knots previously shown in Figure 6.2. The vertices of the graph project to the crossings, and the edges of the graph are

102
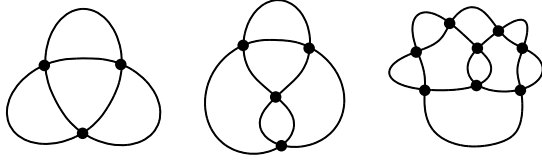
Figure 6.4: The 4-valent graphs obtained from knot diagrams.
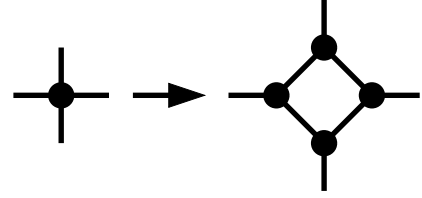
Figure 6.5: Method of vertex replacement to obtain quad graphs.

just the arcs between each pair of crossings. Such a graph is called the *knot universe* or *shadow* by Kauffman [Kau91] and should not be confused with another graph related to knot diagrams that is usually called the knot-graph. Given a DT-code $p(1), p(3), \ldots p(2n-1)$ for an n-crossing knot it is a simple matter to construct the adjacency matrix of its corresponding graph. The first step is to "fill in the blanks" to form the complete DT-code $p(1), p(2), p(3), \ldots p(2n)$ using the relation $p(p(i)) = i$. Since a vertex i in the knot-graph has the two labels i and $p(i)$, it is adjacent to the four vertices $i-1$, $p(i)-1$, $i+1$, $p(i)+1$ (all numbers being taken modulo $2n+1$). This is sufficient to construct the adjacency matrix completely. Also, since this graph corresponds to a knot diagram it is necessarily a planar graph. Therefore obtaining a knot diagram is equivalent to finding an embedding for this graph in the plane.

One further step in the procedure must be done in order to ensure the correct cyclic ordering of edges around each vertex. This requirement can be forced by replacing each vertex in the original graph with a *quad* of vertices as shown in Figure 6.5 to obtain a new graph called the *quad graph* of the knot (diagram).[1] As long as the quad graph is derived from a DT-code of a prime knot in minimal projection, it will be 3-connected (that is, any two vertices may be removed without disconnecting the graph). Here we now see the reason for the DT-code specifying a unique knot diagram. It is related to the fact that 3-connected planar graphs have a unique embedding, up to homeomorphisms of the (extended) plane [Whi33]. We now establish that the quad graphs are indeed 3-connected in the following theorem.

Theorem 1 *Let g be the DT-code of a prime knot K in minimal projection and let G be the*

---

[1]This can be done because we know that the cyclic ordering of the vertices adjacent to vertex i (also known as $p(i)$) is: $i-1$, $p(i)-1$, $i+1$, $p(i)+1$ (up to reversal).

quad-graph obtained from g by the procedure described above. Then G is a 3-connected graph.

Proof

G will be shown to be 3-connected by demonstrating that if any two vertices are removed the graph remains connected. Edges inside quads will be called *quad edges* and edges connecting different quads will be called *principal edges.*

Case 1: The two vertices are adjacent and lie in different quads.

This is the easiest case. Removing the two vertices removes a single principal edge from the graph and is equivalent to "cutting" the string of the corresponding knot diagram. Because the graph is derived from a knot diagram, it must remain connected if these vertices are removed.

Case 2: The two vertices lie in different quads but are not adjacent.

Removal of the two vertices will result in the elimination of two principal edges in the graph. Suppose this separates the graph. Then G must be of the form shown below
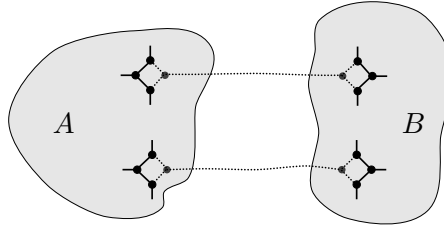
where the dashed edges indicate those removed from G. Either:

(a) both A and B correspond to non-trivial knots. This implies that G was derived from a composite knot, contradicting the hypothesis of the theorem that G is prime.

(b) one of A or B corresponds to the unknot. This implies that G was not obtained from a minimal projection of a knot, again contradicting the hypothesis of the theorem.

Case 3: The two vertices lie in the same quad.

If any two such vertices are removed, then G should remain connected. Something stronger will be shown: that an entire quad may be removed without disconnecting G. Suppose the removal of a given quad separates G into two components A and B. This will delete four principal edges from the graph G. Since G is constructed from a knot diagram it is necessary that two of these

edges go to each component A and B. Therefore G must be of the form
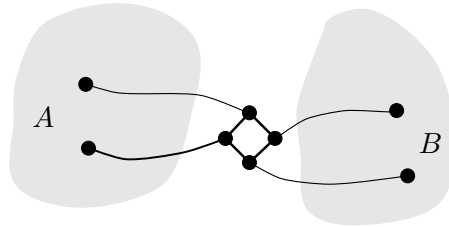
in which case it is evident that K could not have been a minimal projection (because the portion of the knot diagram corresponding to B could be "flipped around" in the plane, eliminating one crossing), violating the hypothesis of the theorem.
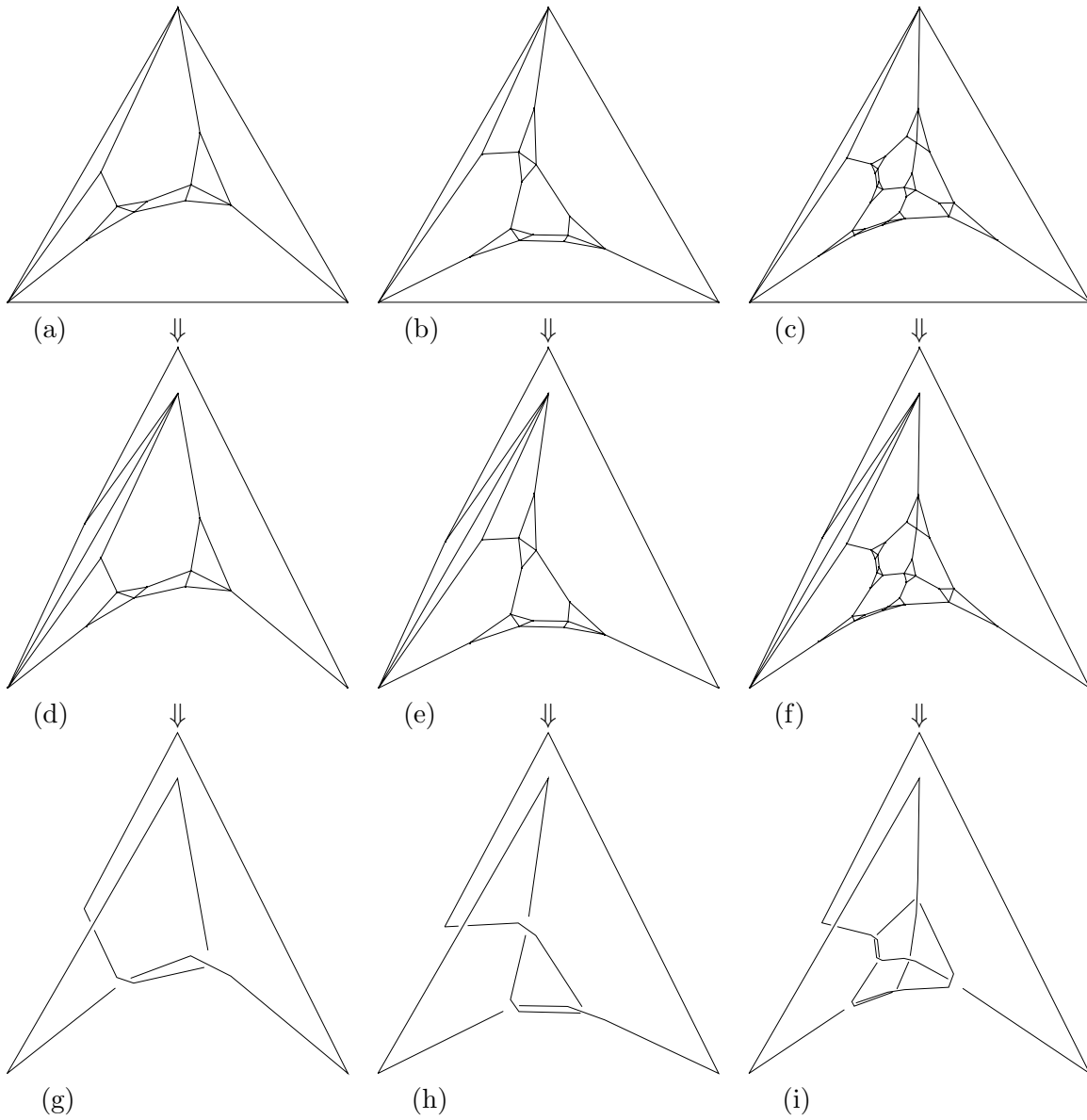
Figure 6.6: Graphs resulting from the embedding process (a–c), after rearrangement (d–f) and their corresponding knot diagrams (g–i). (g) is a diagram of the trefoil, (h) is the figure eight knot, and (i) is the knot 8_{20}.

106

Planar 3-connected graphs enjoy a particularly simple embedding algorithm due to Tutte ([Tut63], see also [Rea79]). This algorithm is easy to describe:

1. Find a face in the graph, say with k vertices.

2. Embed these vertices on a regular k-gon.

3. Embed all other vertices such that each vertex lies at the centroid of its neighbours. This step involves solving a (possibly large) system of linear equations.

It is useful to augment the quad graphs slightly before embedding for the purpose of easily reading off the knot diagram given a diagram of the graph. This is done by subdividing each quad into two triangular faces by adding an extra edge between the pair of opposite vertices that correspond to an overcrossing in the knot diagram. Since adding an extra edge in a graph can only increase its connectedness the embedding algorithm is still valid for these new graphs. One of the triangular faces in the augmented quad graph is chosen and embedded on an equilateral triangle as the first two steps of Tutte's algorithm. For the final step a simple Gauss-Jordan elimination technique [PFTV88] was used. Even though this last phase has an execution time of $\mathcal{O}(n^3)$ it has proven sufficient for the problem sizes tried so far. The result of this embedding is shown for several cases in Figure 6.6 (a–c). To obtain the knot diagram, we first adjust the graphs slightly by adding two extra vertices to get the graphs shown in Figure 6.6 (d–f)[2] from which it is easy to construct the knot diagrams shown in Figure 6.6 (g–i).

It can be seen from the examples shown that the embedding technique does not yield "pleasing" graphs or knot diagrams. In many cases there are too many vertices clustered in one small region. Another deficiency is that crossings in the knot diagrams often cross at small angles, rather than at approximate right angles as seen in most of the knot diagrams in this thesis (taken as being examples of "good" diagrams). One possible fix, although tedious, is to include one extra step in the procedure. An example of this is shown in Figure 6.7 where the quad graph is *relaxed* prior to constructing the knot diagram. This relaxation is done with KnotPlot running in graph-mode. This mode is an (almost) obsolete mode where graphs are

---

[2]Note that these new graphs are no longer 3-connected.
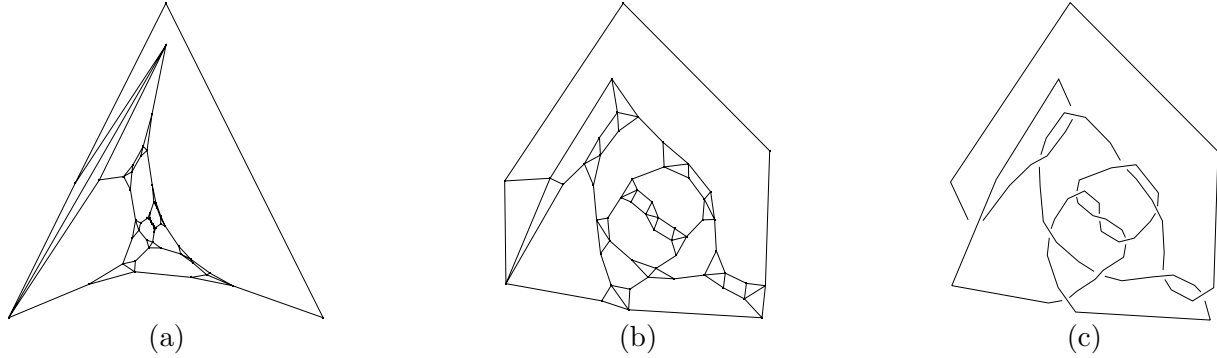
Figure 6.7: Quad graphs and knot diagrams for 13_{1022}. (a) is the "raw" graph obtained from the embedding technique. (b) is after the graph has been relaxed with KnotPlot (with some minor adjustments by hand). (c) is the knot diagram obtained from (b).

represented as adjacency lists. The dynamics is essentially the same as for regular knots (to be described in Chapter 7), except that no collision checking is performed.[3] Because of this, some minor adjustments to eliminate intersecting edges had to be done to arrive at the graph shown in Figure 6.7b. The need to perform adjustments by hand is undesirable and needs to be addressed in future versions of KnotPlot.

The fundamental weakness in this embedding approach is Tutte's algorithm, which simply does not provide adaquate embeddings. It is also prone to numerical instabilities when solving the system of linear equations. Of course there exist better graph embedding algorithms [CON85, Lyo92, FHH+93, DH96], however, none of these methods are specifically concerned with producing good *knot diagrams*. We would like to see, in addition to roughly perpendicular crossings, smooth elegant curves between crossings. Considerations of aesthetic criteria such as these and their relationship to graph embedding is an avenue for further research.

6.2   Relaxing the quad graphs in 3D

A different method for relaxation of quad graphs is to relax the graphs in 3D rather than embedding them directly in 2D. The quad graph is initially embedded in 3D in a random

---

[3]KnotPlot does perform the usual collision checking for the newer implementation of graphs, where graphs are represented as special kinds of knots.

initial random configuration   after relaxation   projected to sphere
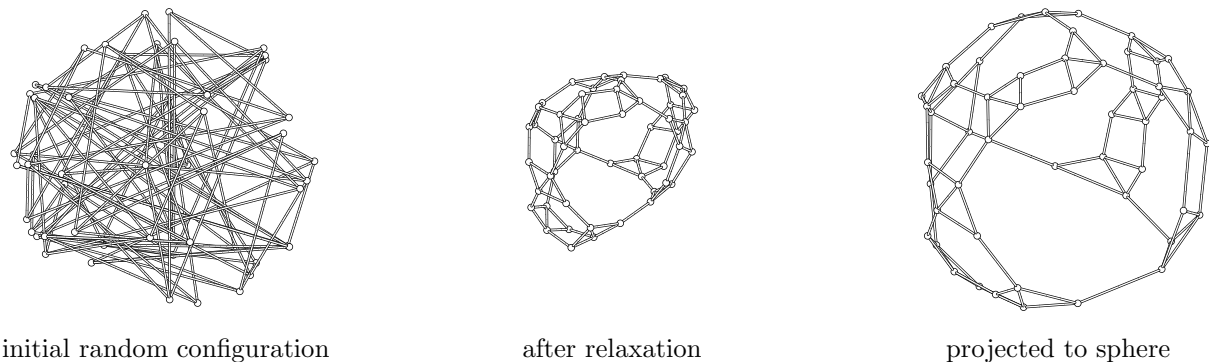
Figure 6.8: Quad-graph for 13_{1022} relaxed in 3D.

configuration. The relaxation of the graph must be done without any collision avoidance. Since the topology of the knot is completely encoded by the graph, this is a valid step; all that is desired is a suitable embedding of the graph in $\mathcal{R}^3$. It has been observed that such graphs naturally relax into configurations that, while not planar, are trivially mapped onto an enclosed sphere (see Figure 6.8). Since the sphere is homeomorphic to the extended plane, it is easy to obtain an embedding in the plane by stereographic projection.

Examples using the 3D method are shown in Figure 6.9. From our experiments, this
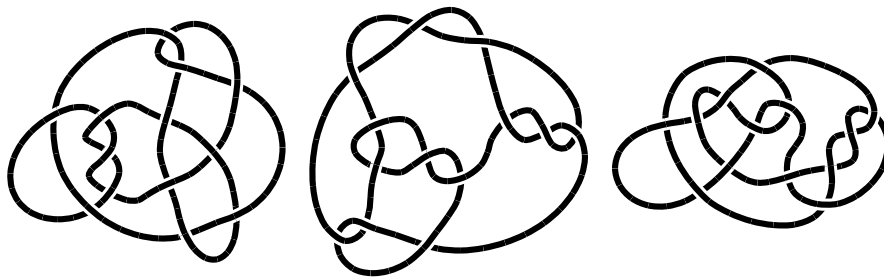
Figure 6.9: Knots 13_{1022}, 13_{2269}, and 13_{4567} relaxed using the 3D technique.

method produces better knot diagrams than would be possible using the method based on Tutte's algorithm. The 3D method is not subject to any of the numerical instabilities of the matrix inversion phase. Another advantage is that the 3D method applies to any quad graph derived from a knot, including composite knots or knots in a non-minimal projection. Figure 6.10 shows an example of the knot diagram obtained from the DT-code of a large composite knot

109
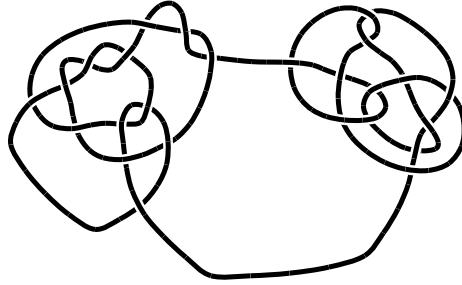
Figure 6.10: Composite knot with DT-code [6 −14 10 20 2 24 18 −4 22 8 12 16 30 36 42 38 46 28 44 48 50 26 32 34 40] relaxed with the 3D method.

(with 25 crossings).[4]

There are also several disadvantages to the 3D method. Although in most cases of interest the quad graphs relax nicely to a configuration that can be projected to a sphere without self-intersections, we have not proven that this will always occur. Experimental evidence indicates that for large knot diagrams (more than 50 crossings), the quad graph will generally get trapped in a local minimum during relaxation in which no projection to an enclosed sphere is possible without self-intersections. Another disadvantage is that the method is somewhat computationally expensive in comparison to the Tutte algorithm. In the cases we have studied, however, this is a small price to pay for obtaining a good knot diagram. Certainly within the range of currently enumerated knots (up to 16 crossings), the technique is perfectly acceptable; Figure 6.11 shows a 36-crossing diagram obtained using the method.

6.2.1 Alternative relaxation scheme

It is tempting to consider knot relaxation entirely by using quad graphs. This would eliminate the need for the computationally expensive collision prevention step needed for the relaxation of knots, the knot type instead being enforced by the topology of the quad graphs. In addition, although the relaxation of the quad graphs uses essentially the same force laws as do the knots, typically fewer vertices are needed to represent a given knot with a quad graph than are needed for the polygonal representation discussed in Chapter 7. Since each relaxation step has a time

---

[4]The reader is reminded that the Dowker code does not necessarily specify an unique composite knot.
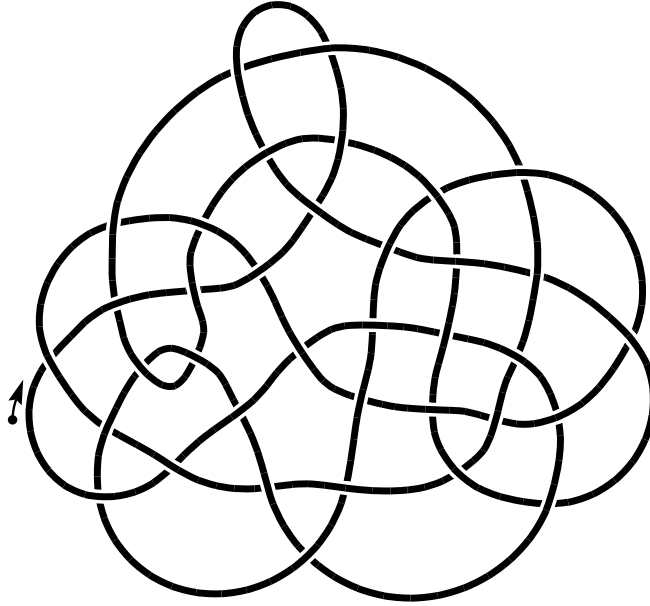
Figure 6.11: 36-crossing knot with DT-code [40 24 10 30 22 52 32 64 46 12 6 42 60 2 8 50 66 16 62 58 28 4 54 34 14 20 68 36 72 26 70 56 48 18 44 38] created with 3D quad method.

complexity quadratic in the number of vertices, this can be a considerable speed advantage. The fact that the quad graphs may not relax to a valid conformation is of no great concern as the final state may be easily checked for self-intersections when projected onto a sphere.

However, it will become clear that the method is no replacement for the techniques to be presented in Chapter 7, simply because the method can never change the number of crossings in the knot; *i.e.* there is no representation for the Reidemeister moves using the quad graphs. Only in the case where one wants to find a good embedding for a given DT-code (especially if that code is known to be from a minimal crossing diagram) is the quad graph method really appropriate. It is an interesting problem for further research whether the method could be enhanced by allowing combinatorial moves on the quad graphs that implement the Reidemeister moves.
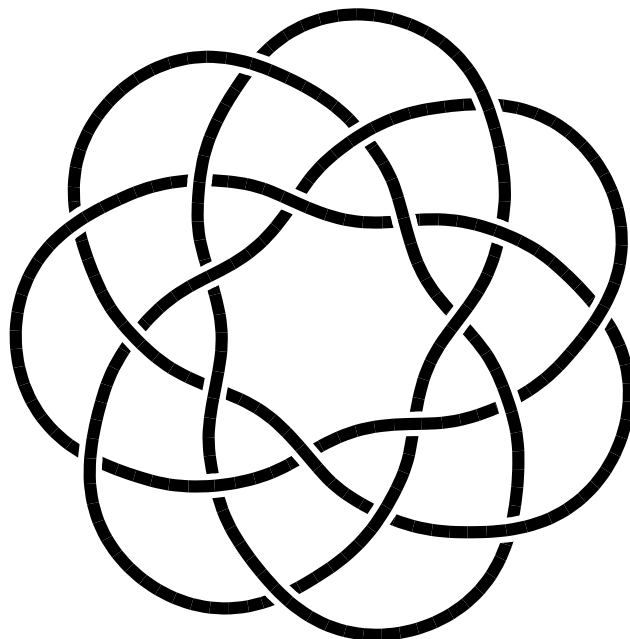
Figure 6.12: "Turk's head" knot generated from a $(4, 7)$-torus knot.

6.2.2 Other applications of the 3D relaxation

This chapter concludes with a less formal example intended to make real knot-tyers[5] happy.
There is a class of knots known as "Turk's head knots". From an inspection of the *Ashley
Book of Knots* [Ash44, chapter 17] it is by no means clear what a rigourous definition of one of
these might be. One subset of the class seems to be alternating versions of torus knots in their
standard form (obtained by a projection of Equation 5.1 to the xy-plane). In other words, the
Turk's head knot and the corresponding torus knot have the same "shadow" on the xy-plane,
except the torus knot is generally non-alternating. Ashley actually gives an algorithm for the
construction (on paper) of such knot diagrams [Ash44, page 243]. Using KnotPlot, it is easy to
do the same. One first computes the DT-code of a chosen torus knot, then eliminates the minus
signs, leaving the DT-code for the corresponding Turk's head knot. The procedure results in the
fine drawing of a Turk's head knot shown in Figure 6.12.[6] This knot has been relaxed slightly

---

[5]This is the officially recognized spelling according to the *International Guild of Knot Tyers*.
[6]This particular knot is also the closure of the braid $(\sigma_1 \sigma_2^{-1} \sigma_3)^7$.

according the methods of Chapter 7 to improve the embedding.

Another possibility using the quad graph approach is to dispense entirely with the two last steps in the procedure, that of projecting to a sphere and the stereographic projection to the plane. This results in the relaxed quad graph shown in Figure 6.13. From this, an embedding
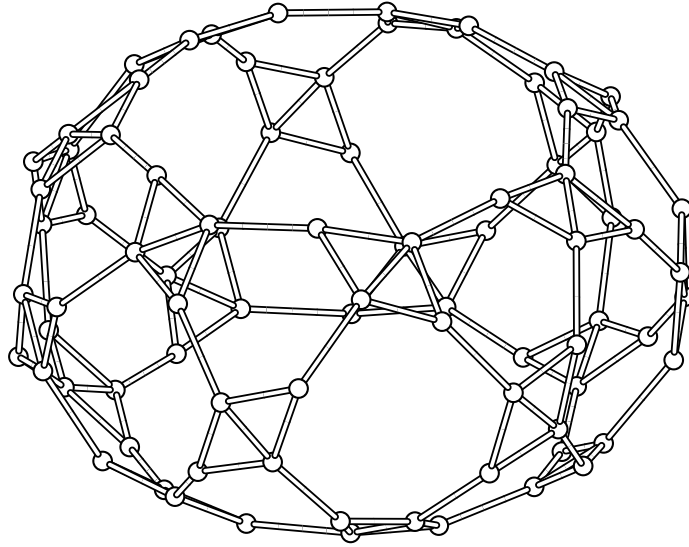
Figure 6.13: Relaxed quad graph for the $(4, 7)$ Turk's head knot.

for the knot could be constructed directly in 3D (this technique is not yet implemented). For one acquainted with the application of Turk's head knots, wrapping about a cylindrical object, the form of the embedding that would result should be familiar.

Chapter 7

Topological refinement

A key point in topological drawing is that users should be able to construct an initial config-
uration for an object while neglecting most issues of geometric placement. This frees them to
concentrate on the important topological features of whatever they are constructing. Because
these initial constructions may be crude in an aesthetic sense, it is often necessary to refine the
configuration. Of course, during this phase the relevant topological features must be preserved.
This chapter discusses methods of *topological refinement*.

One approach to topological refinement is to have most of the work done automatically.
For this, KnotPlot uses a simulated physical model. Objects move according to forces applied,
subject to topological constraints. A second method for refinement is to allow users to inter-
actively modify the embedding, either by applying new forces to influence objects to follow a
desired path, or by performing more drastic editing operations, such as cutting or splicing.

7.1   KnotPlot's dynamical model

KnotPlot uses a simple dynamical model that emulates a pseudo-physical situation. The dy-
namical entities in the simulation are the "beads" or vertices in the knot shown in Figure 7.1
and previously discussed in Section 4.2. It is to these beads that all forces are applied. During
the simulation, each bead simply moves in the direction it is compelled to by the forces that

are applied to it. The "sticks" or edges in the knot are not directly involved in the dynamics, although they provide constraints on the movement of the beads.
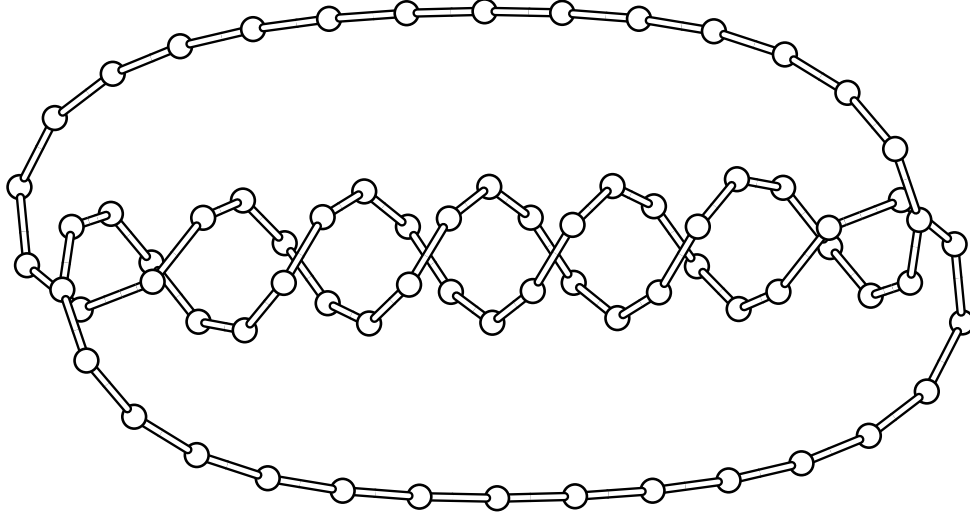
Figure 7.1: A link with Conway number 8 showing the internal "bead and stick" representation.

We have referred to the dynamical simulation as "pseudo-physical" because several modifications to "standard physics" are made to allow more efficient and reliable performance. The first and most important of these is that each bead is moved individually and is constrained to moving a maximum distance $d_{\max}$ during each time step. This permits a simple collision avoidance algorithm to be implemented that ensures preservation of knot-type during the simulation. Secondly, an alternate equation of motion where the applied force is proportional to the velocity ($F \propto v$) is generally used instead of the usual Newtonian dynamics where the force is proportional to the acceleration ($F \propto a$). Thus the beads do not have any inertia; the effect upon the system as a whole is to lose energy. This results in a dynamics similar to one that is always critically damped.

To satisfy "purists", as well as for some applications, KnotPlot can use the strict Newtonian $F \propto a$ dynamics together with an optional velocity-dependent damping force. In normal situations, there is little observable difference between $F \propto a$ with a damping force and the simpler $F \propto v$. The $F \propto a$ dynamics *without* the damping force can be of interest. In this

case, the system conserves energy (approximately, since errors are introduced from clamping the distance that beads can move). We will see later in this chapter examples of knots and links that oscillate about a stable point under the pure $F \propto a$ dynamics (without damping).

One further justification for the $F \propto v$ approach is that in most applications of topological refinement, the user does not care about the path through configuration space followed by the knot. Rather, the final conformation is of greater importance. Informally, one would expect that this final state does not depend on the specific method of integrating the equations of motion, since in either case it is the same forces that come to static equilibrium in the end. This ignores the possibility that some conformations may not be reachable using one of the integration schemes. It implicitly assumes there is a global minimum to the energy state, and relies on local minima being rare if not non-existent.[1]

Before we specify the relaxation algorithm, several definitions are in order.

Definition: *safe position* A (possibly knotted) polygon is in a *safe position* if no two non-adjacent edges in the polygon are closer than a threshold distance d_{close}. This parameter is usually chosen so that $d_{\mathrm{close}} > d_{\mathrm{max}}$.

Definition: *stuck bead* A bead is said to be *stuck* if it cannot be moved to the location specified by the equations of motion without causing the knot as a whole to be moved into a position that is not safe at some point throughout the move.

Definition: *stuck knot* A knot is a *stuck knot* when all of its beads are themselves stuck.

With these definitions, we are ready to specify the basic relaxation algorithm that KnotPlot uses. In the following, the assumption is made that the knot starts out in a safe configuration. At each time step during the simulation, the following sequence of events occurs:

1. sum the forces currently active on each bead

2. for each bead in sequence:

---

[1] At least for prime proper knots. We will see an example later of a link that has a clear local minimum in the energy landscape.

(a) determine the new position of the bead if no topological constraints were active using the total force on that bead and the current physical model for interpretation of that force (*i.e.* $F \propto v$ or $F \propto a$)

(b) if the distance moved is greater than $d_{\max}$, clamp the distance moved at $d_{\max}$ in the same direction

(c) check to determine if moving the bead to its desired location will cause the knot to move into an unsafe position

(d) if the bead can be moved such that the knot is in a safe position, then update the position of the bead — otherwise declare it "stuck" and do not update the position

7.1.1 Force laws

Normally, only two forces are used, an attractive "mechanical" force applied between adjacent beads on the same component and a repulsive "electrical" force applied between all other pairs of beads.

The mechanical force is a generalization of Hooke's law, allowing for an arbitrary power of the distance r between beads,

$$F_m = Hr^{1+\beta} \tag{7.1}$$

where H is a constant and $\beta = 0$ for the standard case of the ideal (linear) spring. The mechanical force is applied only to adjacent beads on the same link component. The electric force also allows for a general power of the distance,

$$F_e = Kr^{-(2+\alpha)} \tag{7.2}$$

where r again is the distance between the two beads. For $\alpha = 0$, the repulsive force falls off as $1/r^2$, just as in the Coulomb case. The electric force is applied to all pairs of beads excluding those pairs consisting of adjacent beads on the same component. In the simulation, the constants H, K, α, and β may be varied. In most of the examples shown in this chapter, $\alpha = 4$ and $\beta = 1$.

For both of these forces, Newton's Third Law[2] is obeyed. The dynamics is similar to

---

[2]Every reaction (force) has an equal and opposite reaction.

that of electro-statically charged masses connected by springs, with the exception that α is usually chosen to be greater than zero (typical values for α range from 4–6). The reason for this has to do with a peculiarity of the $1/r^2$ repulsive force. It will be recalled from classical electrodynamics that two infinitely long charged wires experience no increase in potential if the distance between them is decreased [Jac75]. Although the situation here is quite different, this result from electrodynamics is suggestive that a $1/r^2$ repulsive force will not tend to discourage self-intersections. A higher exponent has proven to be a useful first step at collision avoidance, however, it alone is not sufficient for the task.

7.1.2   Collision Avoidance

For this technique to have any relevance to knot theory, it is imperative that the knot should avoid passing through itself during the relaxation. The problem of collision avoidance is one well known from the field of computer animation, where it is desirable not to have objects pass through one another. Most techniques [MW88] involve fairly expensive calculations to take into account reasonably complex objects such as convex polyhedra. In that work, an attempt is also made to respond to the collision in a physically realistic manner.

In the model considered here, the analysis is simplified by the fact that the knot is represented by a polygonal path in 3D. Therefore the problem of self-intersection reduces to the comparatively trivial problem of finding the distance between two line segments in 3D. In the relaxation algorithm previously specified, the self-intersection problem is solved by requiring that the position of each bead be updated one at a time (no simultaneous updates), and that no bead be moved if the move will result in the knot entering an unsafe position (*i.e.* KnotPlot maintains safeness during relaxation). These conditions, together with the constraint that $d_{\max} < d_{\mathrm{close}}$, are sufficient to prove the following theorem.

Theorem 2 *KnotPlot maintains knot type during relaxation.*

The proof consists of two observations.

Observation 1:   KnotPlot maintains safeness during relaxation. Bead (vertex) positions are updated one at a time.
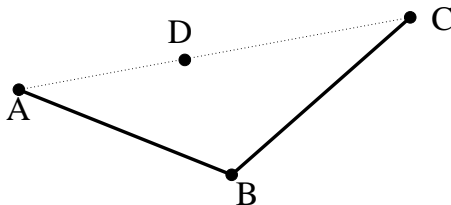
Suppose a bead is about to be moved from position $B \mapsto D$. This move is allowed only if the resulting polygon is in a safe position.

Observation 2: If a vertex in the polygon is moved to a new (safe) position by a distance d where $d < d_{\max} < d_{\mathrm{close}}$, then the knot type remains unchanged.

Proof

Suppose to the contrary that moving a vertex from B to D changes the knot type.
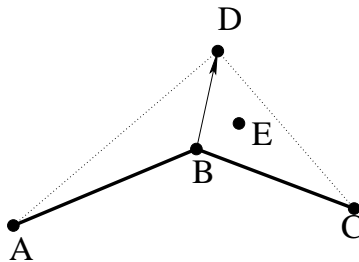
Consider the triangles ABD and BCD (generally non-co-planar). If the knot type is changed by this move, some edge of the knotted polygon must intersect one of these triangles (since otherwise there would be an isotopy between the two positions). Say for argument that this edge intersects the triangle BCD at a point E.
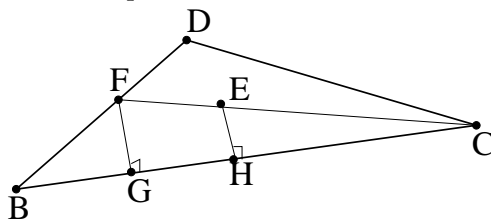
Triangles HCE and GCF are similar, so since $FC \geq EC$ it is true that $FG \geq EH$. By Pythagoras, $BF \geq FG$. Combining this with the fact that $BD \geq BF$ implies that $BD \geq EH$. Now BD corresponds to the distance that the bead has moved. This distance is subjected to the restriction $d_{\max} \geq BD$. Because of observation 1 the segment BC and the segment containing the point E must have been no closer than d_{close} before the move. This means that $EH \geq d_{\mathrm{close}}$. Combining

everything, we have that $d_{\max} \geq d_{\text{close}}$. This contradicts the condition imposed during relaxation, that $d_{\max} < d_{\text{close}}$. Therefore the assumption that the knot type is changed is false.

7.2 Relaxing knots

7.2.1 Results

Now that Theorem 2 has been established, we can proceed to use KnotPlot for relaxing knots with confidence. Figures 7.2 and 7.3 show the relaxation for two simple cases, the initial embeddings

| $T = 0$ | $T = 10$ | $T = 28$ |

| $T = 60$ | $T = 118$ | $T = 223$ |
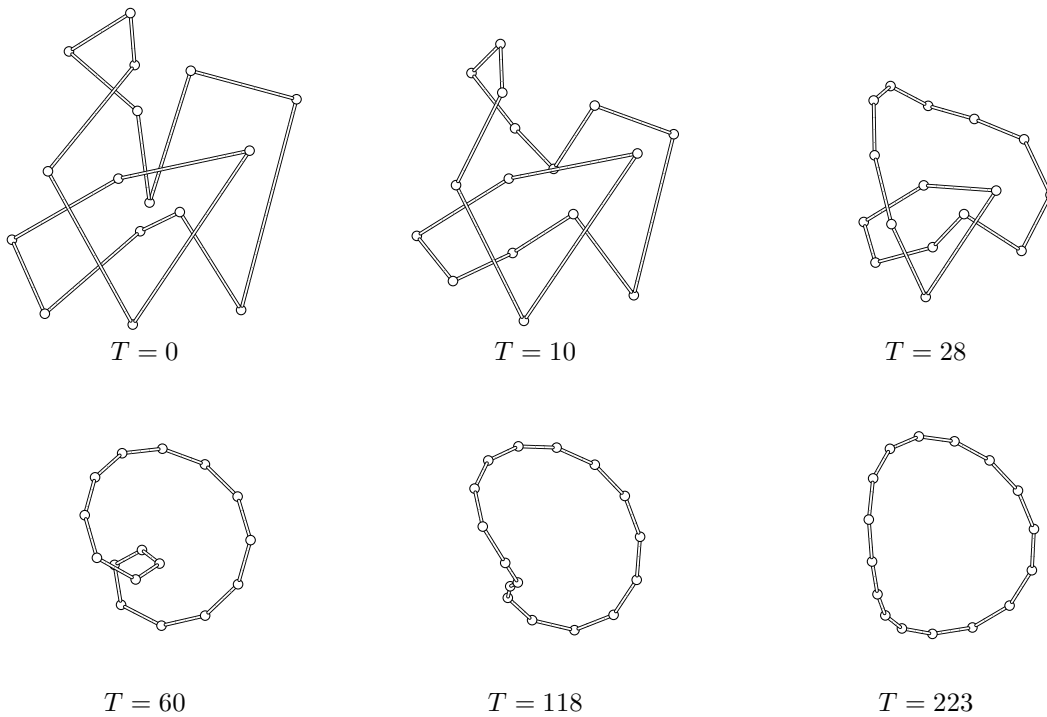
Figure 7.2: Relaxation of a simple unknot

being sketched by hand. Although difficult to illustrate outside of a direct demonstration in front of a computer monitor using KnotPlot, the relaxation algorithm does work to prevent unwanted intersections that might change the knot type. It is easy to see in such a demonstration that some beads become temporarily stuck, becoming unstuck as the balance of forces changes. The visual effect is that the sticks appear to "bounce off" each other.

$T = 0$ $T = 15$ $T = 50$

$T = 129$ $T = 312$ $T = 1022$
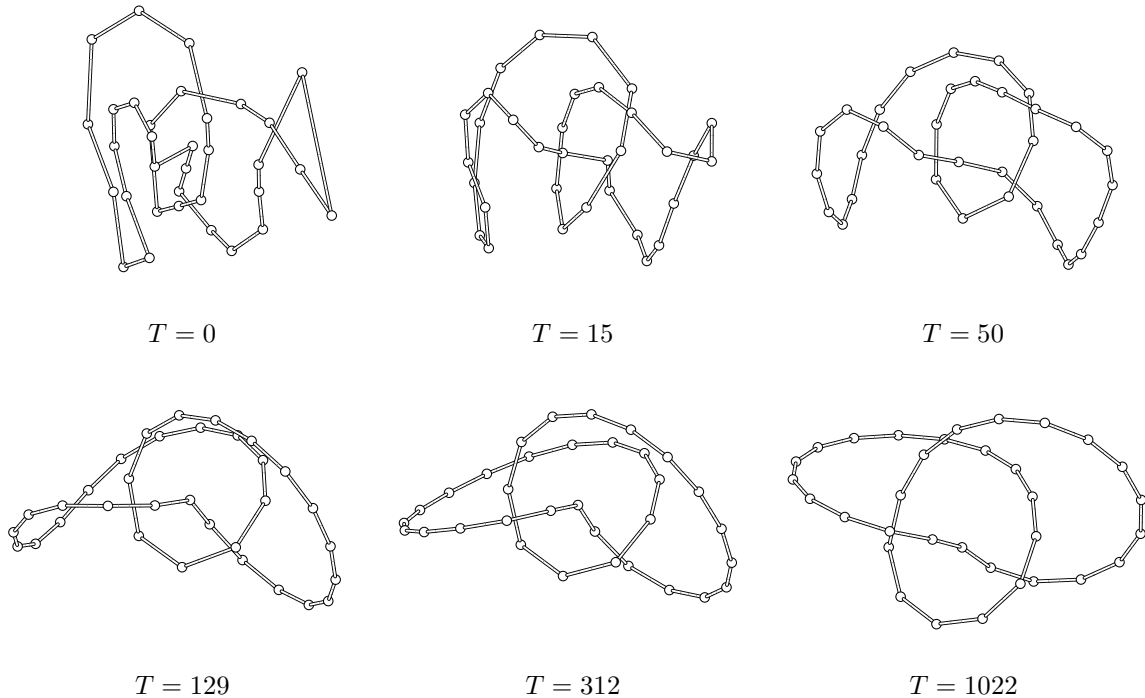
Figure 7.3: Relaxation of a simple trefoil

One interesting[3] observation about make in Figure 7.2 is that the sequence of Reidemeister moves in going from the initial state to the final state are particularly clear (type II from $T = 0$ to $T = 10$, type I from $T = 10$ to $T = 28$, type II from $T = 28$ to $T = 60$, and type I again from $T = 60$ to $T = 118$).[4] It should be emphasized that this results naturally from the simple dynamics; the dynamics itself has no "knowledge" of the Reidemeister moves.

The trefoil is a good illustration of topological refinement. The initial construction is rather "messy" (drawn quickly by hand), and is not even a minimal projection. As the relaxation proceeds, the trefoil approaches the canonical form for the trefoil discussed in Section 3.5. Achievement of this canonical form is more clearly shown in Figure 7.4, also a trefoil but this time the initial configuration is derived from the Conway notation. Here we see the canonical

---

[3]One reviewer pointed out that this is not interesting at all, as Reidemeister's theorem says one must see such a sequence in any isotopy. Of course this is true, however, I feel it is worth mentioning here to make it perfectly clear that the Reidemeister moves are not "hard-coded" into KnotPlot in some way, an impression that some people still have even after the dynamics have been explained.

[4]The symbol T represents the number of iteration steps in the relaxation algorithm.
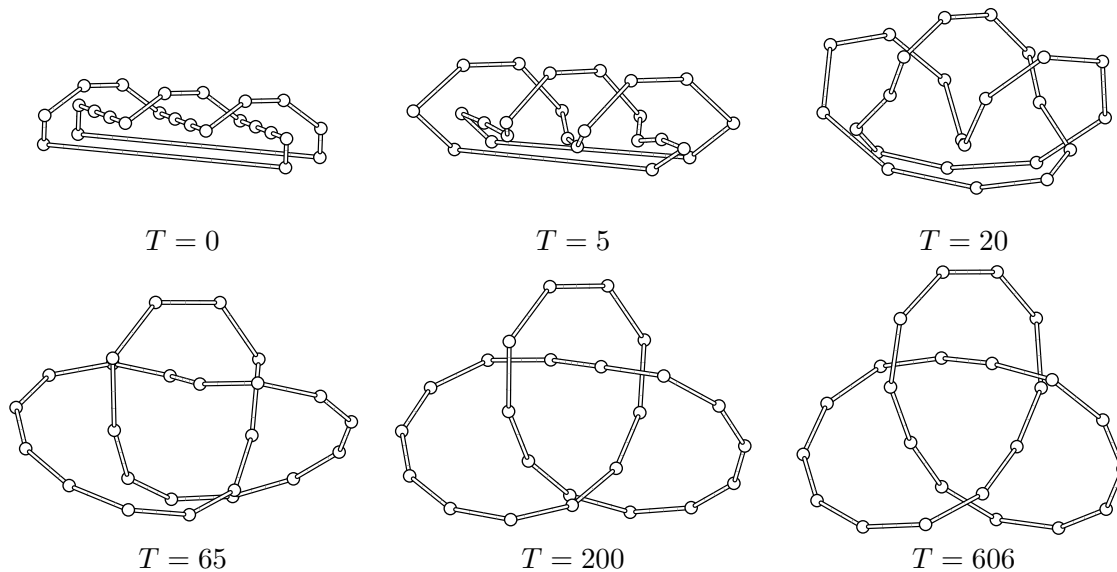
Figure 7.4: Relaxation of a trefoil starting from Conway notation 3.

form emerges from the initially somewhat arbitrary embedding.

The relaxation examples shown so far are rather trivial; *i.e.* the knot type in each case is easily recognizable by inspection. A more interesting test of the technique is starting with one of the two "difficult" unknots seen previously in Chapters 1 and 3, the "monster" unknot and the Freedman unknot. Figure 7.5 shows the relaxation of the monster unknot. The initial configuration at $T = 0$ has no simplifying Reidemeister moves (type I or II), so the knot must be made more complicated (in terms of number of crossings) before it can be simplified. At $T = 741$, we see that the knot has reached a maximum of 12 crossings, after which it begins to simplify.

In the case of the Freedman unknot the relaxation also proceeds at first to make the knot more complicated (Figure 7.6). Our relaxation is qualitatively similar to Kusner and Sullivan's relaxation of the same knot [KS94] (they started from a slightly different initial configuration). Indeed, the KnotPlot relaxation is somewhat more "satisfying" in some ways; not exhibiting the "tight spots" shown in [KS94]. These tight spots in [KS94] are largely due to the Möbius invariance of the energy model they used, which effectively "doesn't see" such features. In
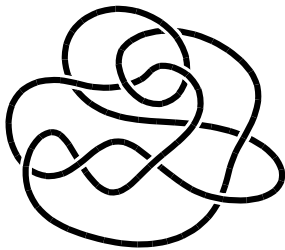
$T = 0, \ E_{\mathrm{MD}} = 335$
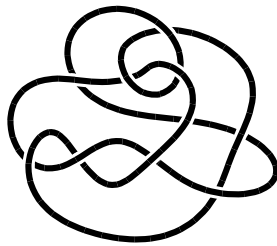
$T = 17, \ E_{\mathrm{MD}} = 304$
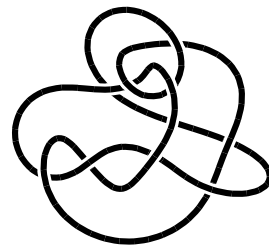
$T = 220, \ E_{\mathrm{MD}} = 277$
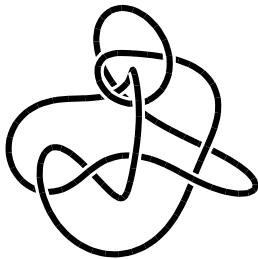
$T = 741, \ E_{\mathrm{MD}} = 253$
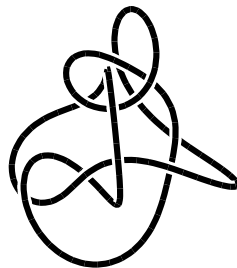
$T = 1756, \ E_{\mathrm{MD}} = 228$
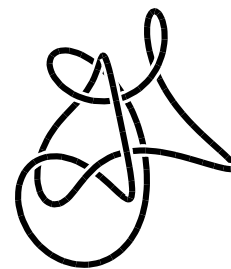
$T = 3144, \ E_{\mathrm{MD}} = 203$
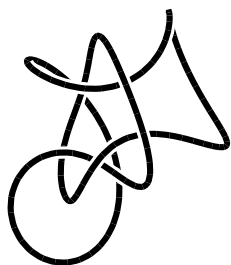
$T = 5514, \ E_{\mathrm{MD}} = 176$
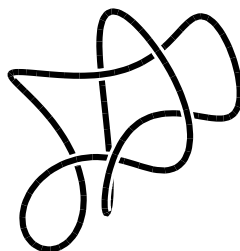
$T = 13310, \ E_{\mathrm{MD}} = 144$
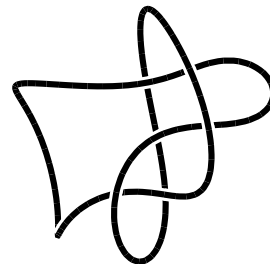
$T = 31121, \ E_{\mathrm{MD}} = 116$
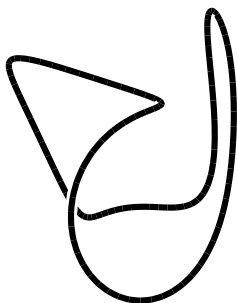
$T = 54518, \ E_{\mathrm{MD}} = 70$
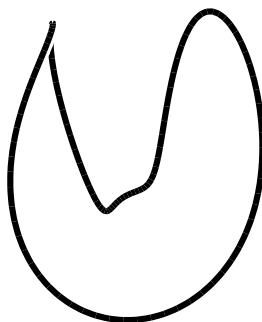
$T = 80000, \ E_{\mathrm{MD}} = 49$
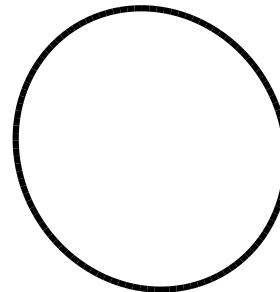
$T = \infty, \ E_{\mathrm{MD}} = 10$

Figure 7.5: Relaxation of the "monster" unknot, shown at approximately equal steps in E_{MD}.

$T = 0, \ E_{\mathrm{MD}} = 1217$ $T = 23, \ E_{\mathrm{MD}} = 1132$ $T = 226, \ E_{\mathrm{MD}} = 1048$

$T = 5460, \ E_{\mathrm{MD}} = 963$ $T = 6800, \ E_{\mathrm{MD}} = 877$ $T = 9860, \ E_{\mathrm{MD}} = 794$

$T = 12880, \ E_{\mathrm{MD}} = 708$ $T = 13360, \ E_{\mathrm{MD}} = 624$ $T = 14420, \ E_{\mathrm{MD}} = 539$

$T = 15360, \ E_{\mathrm{MD}} = 455$ $T = 21180, \ E_{\mathrm{MD}} = 371$ $T = \infty, \ E_{\mathrm{MD}} = 286$
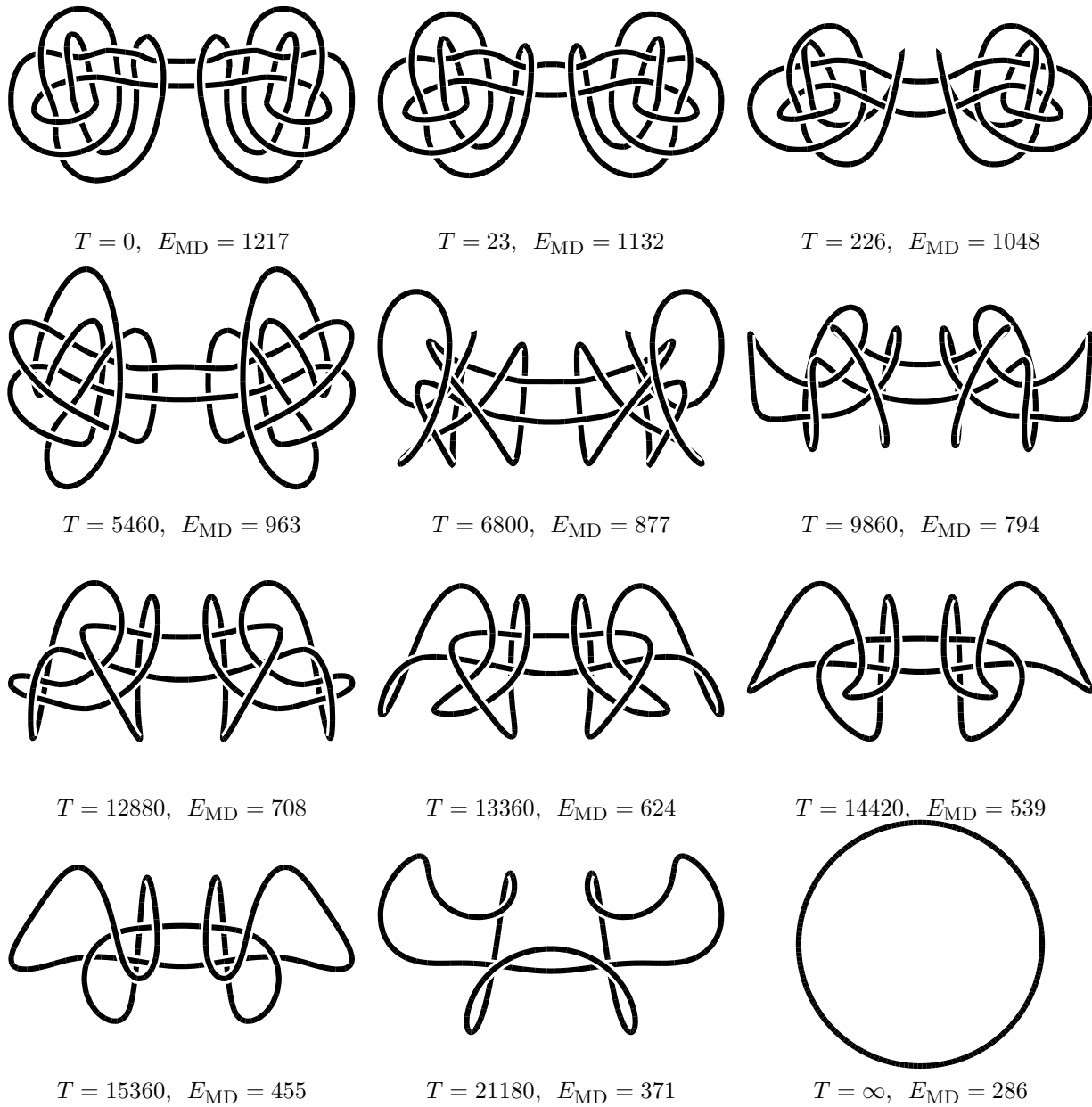
Figure 7.6: Relaxation of the "Freedman" unknot, shown at approximately equal steps in E_{MD}.

124

addition, the greater "openness" in the KnotPlot case is a result of the steep falloff (high power) of the repulsive force used.

Finally, this section would not be complete without showing the relaxation of at least one non-trivial knot. Figure 7.7 shows the relaxation of the torus knot 5_1 starting from a tangled initial configuration. It is interesting to note that the final state does *not* have the expected symmetry that the same knot ($K_{2,5}$) shows in Colour Plate 1. Similar results are shown in [KS94] for this knot; it appears that a local minimum has been found in both cases.

7.2.2 Limitations of the dynamics

Although the relaxation method appears to work well in the examples given so far, there are several problems and limitations of the dynamics. One drawback is that the operation of collision checking and the repulsive force are both of $\mathcal{O}(N^2)$ in the number of beads, N. Also each collision check is potentially quite expensive. Specifically, determining the distance between line segments in 3D, although straight-forward, involves a considerable amount of vector algebra. However, it is not necessary to know the distance between each line segment, but simply whether two line segments are farther apart than some threshold distance. A useful first step is then to perform a bounding box check. Only if this fails is the more expensive routine that determines the distance between the line segments called. This gains a factor of about five in the speed of the relaxations.

The method of relaxation described above works well in a large number of cases. All the knots contained in KnotPlot's on-line catalogue were relaxed using the force laws as presented. Most of these were sketched by hand, although many originated from a Conway notation description. It was found in examining the logs of the simulations that generally the force laws alone would have been sufficient to prevent self-intersections. However, the explicit collision prevention was needed in a few cases (less than 2%). Of course the times that it is needed more than justify the extra expense in performing the collision checking step.

One problem observed with the dynamical simulation is that on occasion a knot will get stuck in an "awkward" orientation. This is partly due to that fact that the force law works to encourage collisions in some cases. Figure 7.8 illustrates a typical situation where this occurs.

$T = 0, \ E_{\mathrm{MD}} = 381$ $T = 18, \ E_{\mathrm{MD}} = 368$ $T = 30, \ E_{\mathrm{MD}} = 354$ $T = 39, \ E_{\mathrm{MD}} = 342$

$T = 60, \ E_{\mathrm{MD}} = 329$ $T = 120, \ E_{\mathrm{MD}} = 315$ $T = 209, \ E_{\mathrm{MD}} = 302$ $T = 307, \ E_{\mathrm{MD}} = 289$

$T = 401, \ E_{\mathrm{MD}} = 276$ $T = 551, \ E_{\mathrm{MD}} = 262$ $T = 781, \ E_{\mathrm{MD}} = 249$ $T = 1081, \ E_{\mathrm{MD}} = 236$

$T = 1481, \ E_{\mathrm{MD}} = 222$ $T = 2181, \ E_{\mathrm{MD}} = 208$ $T = 3381, \ E_{\mathrm{MD}} = 196$ $T = 21381, \ E_{\mathrm{MD}} = 183$
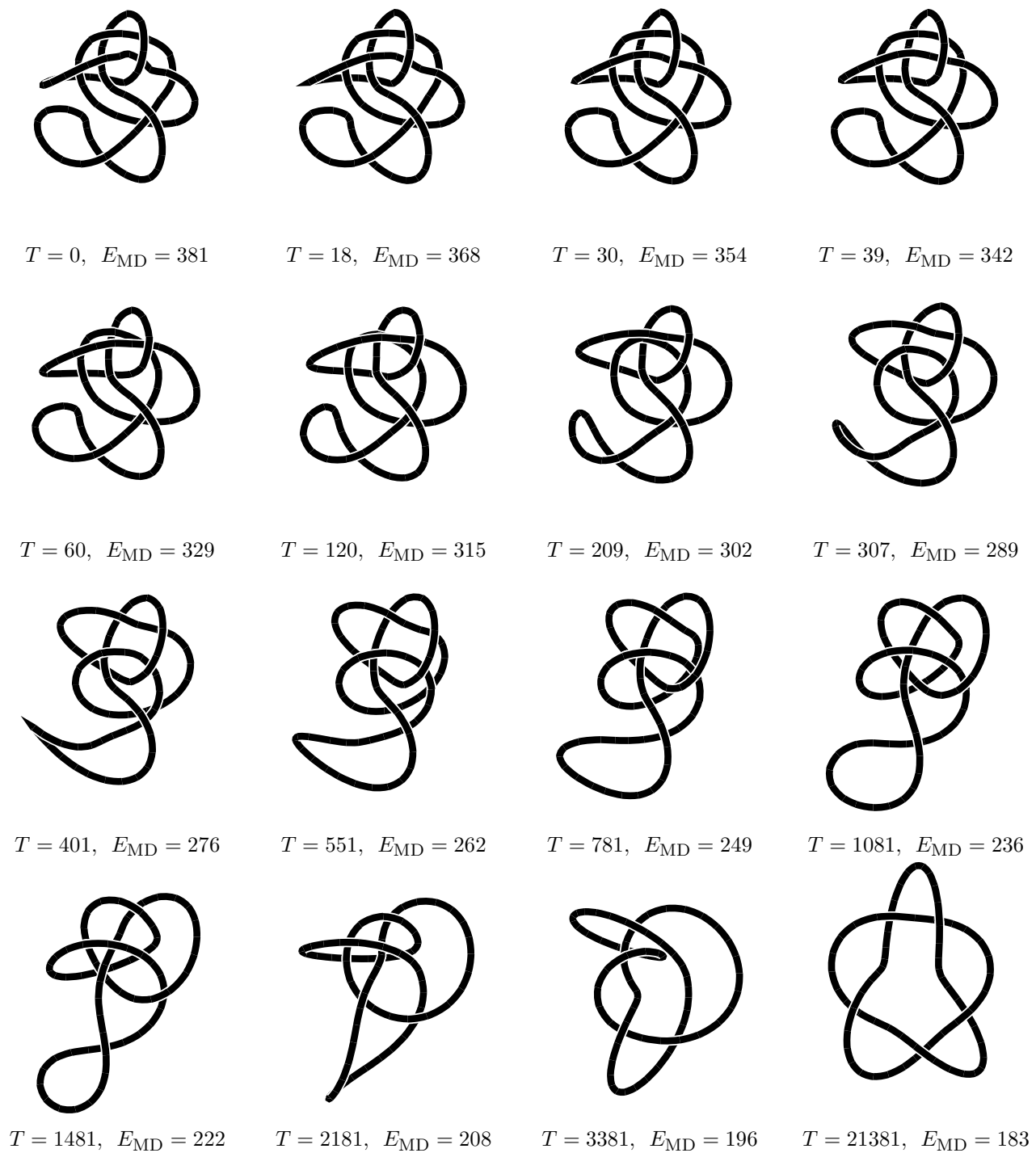
Figure 7.7: Relaxation of the 5_1 knot, shown at approximately equal steps in E_{MD}.

$T = 0$ $\quad\quad\quad\quad$ $T = 18$ $\quad\quad\quad\quad$ $T = 27$ $\quad\quad\quad\quad$ $T = 33$
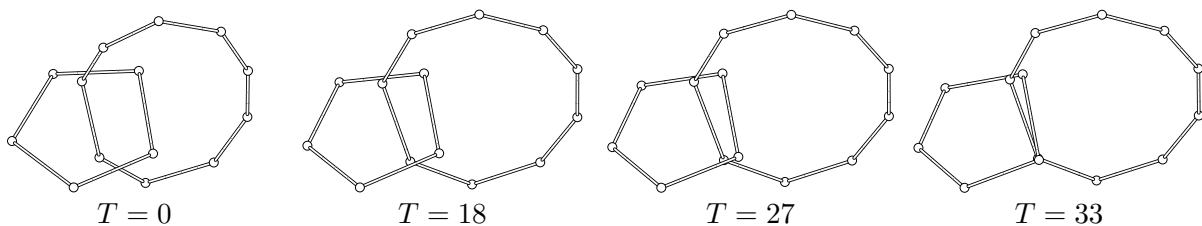
Figure 7.8: Force laws encouraging collision.

The two right most beads in the component with the fewer beads have a net force in the left direction of a slightly larger magnitude than the neighbouring beads in the other component at $T = 0$. As the simulation proceeds, the components slowly drift closer to each other until at $T = 33$ two beads in each component become stuck. The collision shown in Figure 7.8 occurs when using a (default) value of $\alpha = 4$ in Equation 7.2. With $\alpha = 6$ the collision does not take place, however, increasing the parameter α is not normally desirable as this can lead to numerical instability. A better solution is to increase the number of beads in the knot.

Occasionally the entire knot becomes stuck as seen in Figure 7.9 which shows a stuck unknot. Again, problems of this sort can be avoided simply by using a sufficient number of beads in the knot model. Curiously, this knot becomes unstuck and eventually relaxes to a regular
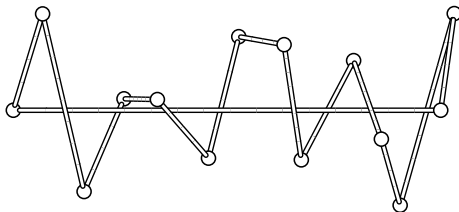
Figure 7.9: Stuck unknot for $\alpha = 4$.

polygon if α is reduced to zero. Normally using $\alpha = 0$ would lead to many collisions. Since the parameters of the simulation can be modified interactively, it is often useful to do so to get out of stuck situations. However, this method requires a considerable about of experience with KnotPlot, and is not guaranteed to always succeed.

127

7.3   Extensions to the dynamical model

Although the method of relaxation so far presented works well in many cases, it is interesting to consider extensions to the dynamical model. These extensions may be for the purpose of studying new phenomena in knot relaxation, or to attempt to deal with the situations where the relaxation method fails. In the latter case, the previous section suggested one technique (interactive modification of parameters) that could be used, however, it is desirable to have available more reliable methods that don't necessarily rely on user interaction.

The extensions discussed in this section are of three types. First of all, there are new forces, one of which is interactive. Secondly, there are methods to *constrain* motion. Finally, there are techniques that step outside purely physical models for knot dynamics. These methods change the geometry of the knot by altering the polygonal representation (while preserving the knot type).

7.3.1   Forces

KnotPlot has more than a dozen force models available for use by experimenters. Section 7.1.1 discussed the two most commonly used forces, the mechanical and electrical force. Both of these are parameterized families of forces; each has a relative magnitude and an associated exponent. The mechanical force is a generalized spring that can vary from being strictly linear (as are most real-world springs) to highly non-linear. The repulsive electrical force has one extreme where it behaves much like electrical charge (force falls off with the inverse square of distance), and another extreme (the normal setting) where the force falls off with a higher power of the distance, thus behaving as an effectively short-range force.

Even with these two forces, the number of possible parameter settings is large. We have found that useful behaviour results from a relatively narrow range of parameter settings. The addition of still more force laws, each with its set of parameters might compound the problem of "floundering in parameter space", however, we find that by implementing forces that have simple physical interpretations, we can rely on the intuition of the user to aid in finding useful combinations and parameter settings.
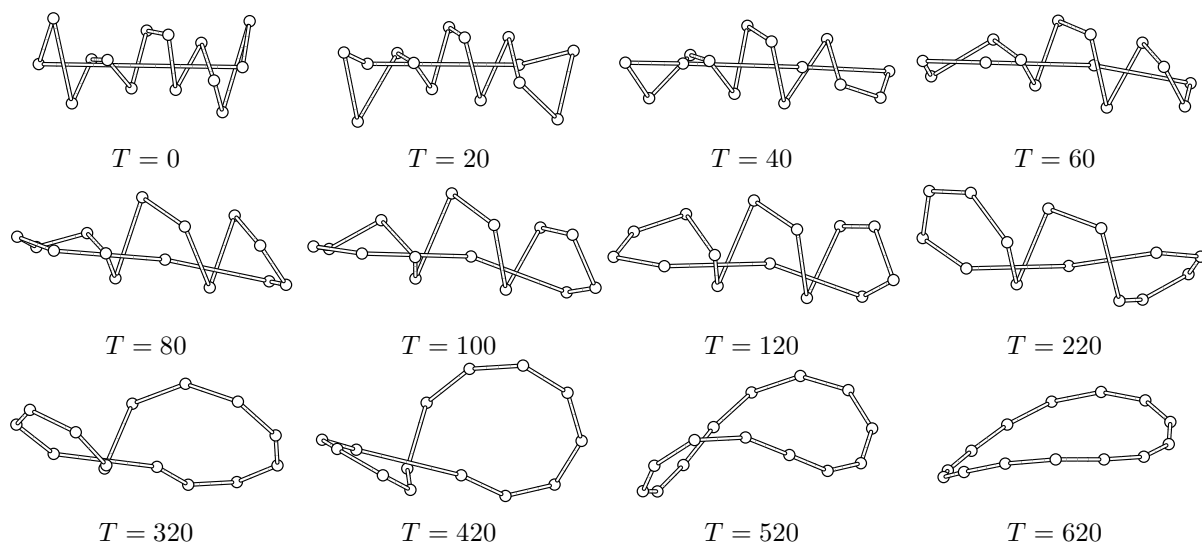
Figure 7.10: Stuck unknot becoming unstuck with thermal force. The thermal force was used for only the first 120 iterations.

In order not to overburden this section with an exhaustive discussion of each of the new forces, we will concentrate on the few that have found useful applications in the research presented in Chapter 8. Other forces, such as gravity, are more of a curiosity, but may be useful for users who are seeking to design something out of the ordinary. As with other details about KnotPlot, the manual [Sch97b] has full information.

Thermal noise

The "thermal noise" force is the simplest of the new forces. It is not really "thermal" in a strict physical sense, however, qualitatively the effect is similar to thermal agitation. If this force is enabled, then during each time step every bead in the knot receives an additional force with direction uniformly distributed on the sphere and magnitude uniformly distributed between zero and a user-selectable maximum.

The thermal force is quite good at getting out of stuck situations. Figure 7.10 shows the stuck unknot of Figure 7.9 becoming unstuck after a little bit of relaxation with the thermal force enabled. Sometimes it is useful to run the knot dynamics using *only* the thermal force.

Chapter 8 discusses an application of this method in finding stick numbers of knots.

Push/pull rockets

All of the forces so far described work automatically, with the exception that the user can inter-actively modify whether the force is enabled (by turning it on or off) as well as the parameters associated with the force. However, with the use of *push/pull rockets* the user is able to decide *interactively* where and when to apply the force. This force works *only* in an interactive fashion and is applied to only one bead in the knot. The experimenter chooses this bead by running the simulation and clicking with a mouse button on a bead in the knot. Depending on which mouse button is pressed, a force is applied to the chosen bead along the line of sight either in the direction of the viewpoint, or in the opposite direction. The magnitude of this force is a user-selectable parameter, it is usually chosen so that the force overwhelms all other forces acting on the bead. The effect is akin to attaching a "rocket" to one part of the knot that pushes or pulls along the line of sight. A typical interaction technique is to apply the rockets intermittently to sections of the knot that appear stuck, while rotating the knot with a virtual trackball. Normally this would be done with the two default forces (mechanical and electrical) active, however the push/pull rockets may be used alone. In the latter case the experimenter can directly manipulate the knot (one bead at a time) while ensuring that the knot-type remains unchanged.
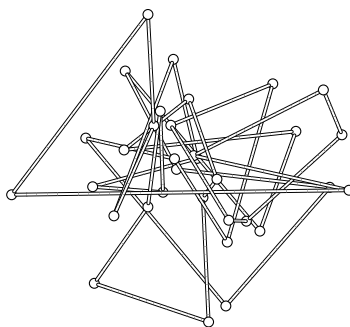
Figure 7.11: Stuck trefoil that can be simplified using push/pull rockets.

Push/pull rockets are typically more effective at getting out of stuck situations than the

thermal force. Figure 7.11 shows a badly stuck trefoil: two out of three beads in this knot become stuck if the dynamics is run using only the two default forces. By using these forces in conjunction with the rockets (and rotations of the knot), it is easy (after some practice) to manipulate the knot until it achieves the canonical form for the trefoil. "Topological puzzles" such as shown in Figure 7.11 are examples of *computational steering*, where the experimenter affects the outcome of a computation by directly intervening and steering the computation to a desired goal, based upon information gathered through visualization.

7.3.2 Constraining motion

Bead masses

In the examples shown so far, the beads in the knots can be considered *identical*, in particular, they all have the same mass. Therefore with a given force, each bead will respond in the same way (ignoring any topological constraints upon the movement). A simple generalization of the dynamics is to lift this restriction, and allow arbitrary masses for beads. Depending on which form of dynamics is used, the bead mass m will provide a scale factor on the amount of movement on each time step (using the $F \propto v$ interpretation, now $F = mv$), or the velocity change on each time step (using $F = ma$).

So far, we have not found any especially interesting applications of this feature, with the exception of the next simplest case where beads of two different masses are used: the default (unit) mass and infinite mass. Beads of infinite mass are *fixed* in location. This can be useful for drawing specific knot diagrams as shown in Figure 7.12. The trefoil and the endpoints of the open component are held fixed using beads of infinite mass, whereas other beads in the open component have the default mass and so are free to move in the normal manner. If the trefoil were not held fixed during relaxation, it would tend to deform due to the presence of the other component, which in turn would tend to unravel and eventually become free of the trefoil. In this example, we might be using topological drawing to illustrate a particular idea (perhaps related to the computation of the knot group, for example) and it is desirable to have the trefoil retain its canonical form.

initial construction   after relaxation   drawn as smooth
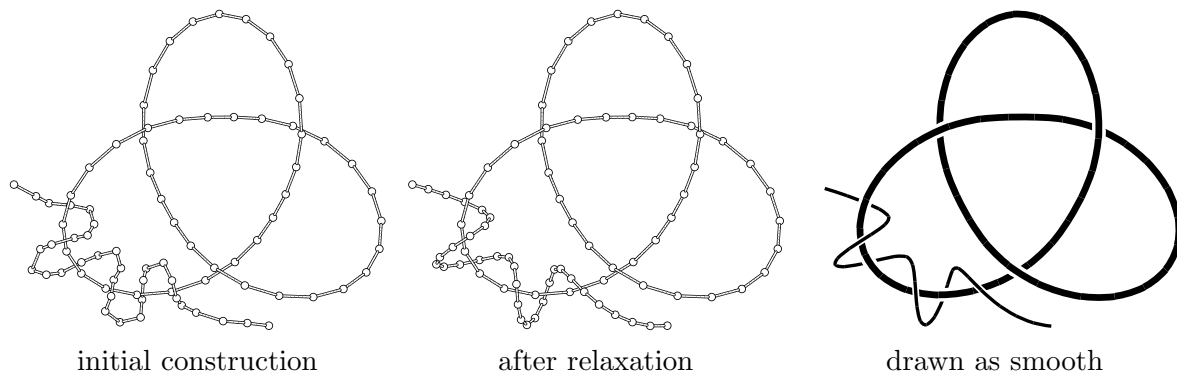
Figure 7.12: Relaxation using beads of different masses.

Anchors

A second method of constraining movement is through the use of *anchors*. Anchors are points in $\mathcal{R}^3$ associated with each bead in the knot. Each bead is attached to its anchor point using a spring with a force law similar in form to Equation 7.1. The effect of anchors is that, once they are set, beads are not free to drift "too far" from their anchor points. This is useful in

initial sketch   after relaxation   drawn as smooth
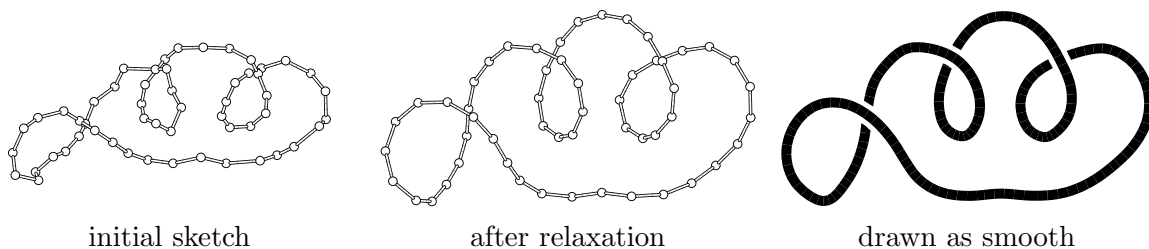
Figure 7.13: Relaxation using anchors to improve a sketched knot.

improving rough sketches, such as shown in Figure 7.13. On the left is shown a configuration sketched by hand. After this initial rough sketch, the anchor points are set to be equal to the bead positions and the dynamical simulation is run with the anchors and the default mechanical and electrical forces. It can be seen that after the "knot" has come to an equilibrium position (middle figure), the salient feature of the initial sketch (showing three loops) is preserved but the embedding has been improved. Without anchors, of course, this unknot would simplify to a round circle eventually.

      Anchors have been used in the manner described above in many cases for the production

initial configuration   after 1 step   after 8 steps   after 25 steps
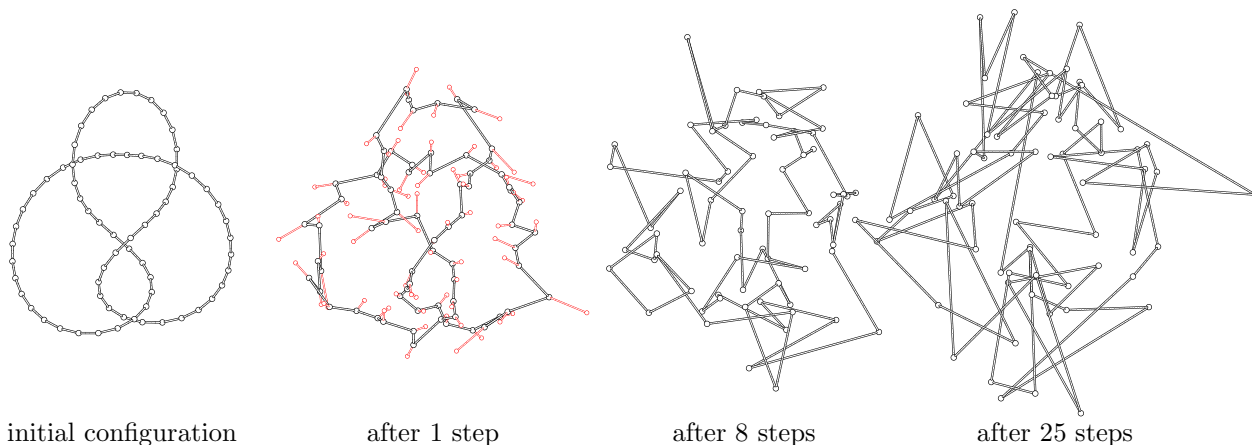
Figure 7.14: Creating a random embedding of a figure-eight knot using anchors.

of knots in the knot catalogue, especially for knots that are unique in some way. For example, the knots shown in Figure 4.1 and the two decorative knots shown in Figure 5.2 were originally sketched by hand (using a scanned image as a backdrop) and then refined using relaxation with the anchor points set to the initial bead locations.

A quite different application of anchors is in the creation of random embeddings of specific knot-types. While KnotPlot has several methods of creating random knots (Section 5.1.2), given knot-types are created only by chance; one might have to generate many knots before one with a desired knot-type results. Figure 7.14 shows how an initial embedding of a figure-eight knot may be randomized through the use of anchors. In each randomization step, the following sequence of sub-steps is taken:

(a) Save the current configuration of the knot.

(b) Randomly displace each bead in the knot, without regard to preserving knot-type. It is not important to preserve knot-type since these locations will only be used to set the anchor locations.

(c) Set the anchor locations to the current bead locations.

(d) Restore the configuration of the knot from the position saved in sub-step (a).

133

(e) Relax the knot with anchors enabled, and possibly other forces as well. In the example shown in Figure 7.14, no other forces are used.

A typical result after one complete randomization step is shown as the second diagram in Figure 7.14. The anchor locations set during sub-step (b) of step 1 are shown in a lighter colour. In this example, the anchor locations are relatively close to the bead locations. Of course the anchors could be set to arbitrary positions during sub-step (b). Whatever the locations of the anchor points are, each bead will be attracted towards its anchor point during sub-step (e), possibly becoming stuck along the way. The net effect of many such randomization steps is to incrementally distort the knot to a new random configuration, while preserving the knot-type.

7.3.3   Bead tricks

One of the most useful techniques in knot relaxation or recognition is to "step outside of physics". Since our domain is not molecular dynamics but topology, we are free to explicitly modify the polygonal knot by adding or removing vertices, as long as these procedures maintain the knot-type. Huang, Kauffman, and Grzeszczuk [GHK97] (independently) arrived at the same conclusion that this method greatly improves the efficiency of the relaxation procedure. Our method is almost identical to theirs, except that they always ensure that the number of vertices in the knot is unchanged as the relaxation proceeds; *i.e.* the method is essentially a vertex shift operation. They impose this constraint in order to maintain a greater physical realism, although strictly speaking even a vertex shift while maintaining the number of vertices is somewhat non-physical.

KnotPlot doesn't impose any constraint on keeping the number of vertices in a knot constant. If this is desired, then the deleting and adding of vertices can be done in a single step. In the examples we show in this thesis, this has not been done.

Deleting beads

Figure 7.15 illustrates the bead deletion procedure. First a bead is chosen for deletion, this is often done via a random selection. Then an attempt is made to move the bead to a target

location that lies along the line segment joining the two neighbouring beads. Usually this is simply the midpoint of the line segment. The bead is moved incrementally, in steps such that
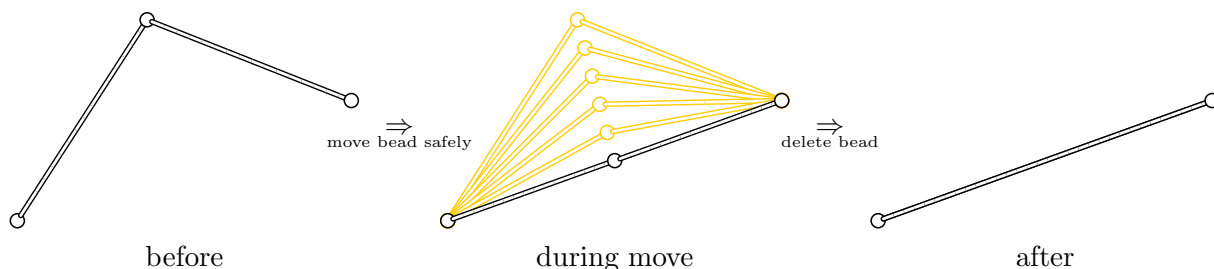
Figure 7.15: Bead deletion procedure.

the distance moved during each step is no greater than $d_{\max}$. After each step, the knot position is checked for safeness. If the knot is in an unsafe position after any step, the deletion procedure is aborted and the chosen bead returned to its original location. If the knot is in a safe position after the bead reaches its target location, the bead is deleted. Because the manner in which the bead is moved mimics the movement that might occur using the normal dynamics, Theorem 2 ensures that the knot-type remains unchanged.

Although this method is computationally equivalent to the method used by Huang, Kauffman, and Grzeszczuk [GHK97], which checks to see if any part of the knot intersects the triangle formed by the bead chosen for deletion and its neighbours, our method has several useful features. First of all, drawing the knot at each step during the bead move produces an interesting animation of the algorithm. Normally, for efficiency reasons, drawing is turned off during the procedure. Secondly, the implementation allows for *general* move operations. Examples would be moving the bead to the nearest location with integer coordinates or perhaps moving the bead along a (user) specified path. Placing these move operations within a unified framework based on the same collision prevention algorithm as the normal dynamical model is conceptually useful.

Splitting edges

The companion procedure to deleting beads is to add extra beads by splitting an edge. This operation trivially sub-divides an edge into two edges of equal length. The only caveat is that

135

splitting an edge may cause the knot to no longer be in a safe position, even though splitting an edge doesn't change knot-type. Because Theorem 2 only ensures the knot-type remains unchanged in subsequent relaxations if the knot is in a safe position, splitting is not allowed if safeness would be lost.

Splitting edges becomes more interesting if it is done *automatically*. A useful technique is to check for stuck beads after a relaxation step, and then to split the associated edges in the knot (while maintaining safeness). For efficiency reasons, we have found it best to perform the check and splitting procedure every M relaxation steps, where $M \approx 10$. Auto-splitting in this way is interesting for two applications. The first is in finding a *natural length* for different knot-types. For this we start with a reasonable initial configuration for the knot, based on some natural metric. A good choice is a minimal-stick candidate in a minimal energy configuration for the minimum distance energy (see Chapter 8 to see how these are found), but many other choices would give similar results. The next step is to run the normal dynamics using the two default forces with auto-splitting enabled. During the early stages of the relaxation, many beads become

3_1 length $= 23.4$ 6_3 length $= 54.1$ 9_{42} length $= 56.8$
number of beads $= 15$ number of beads $= 36$ number of beads $= 36$
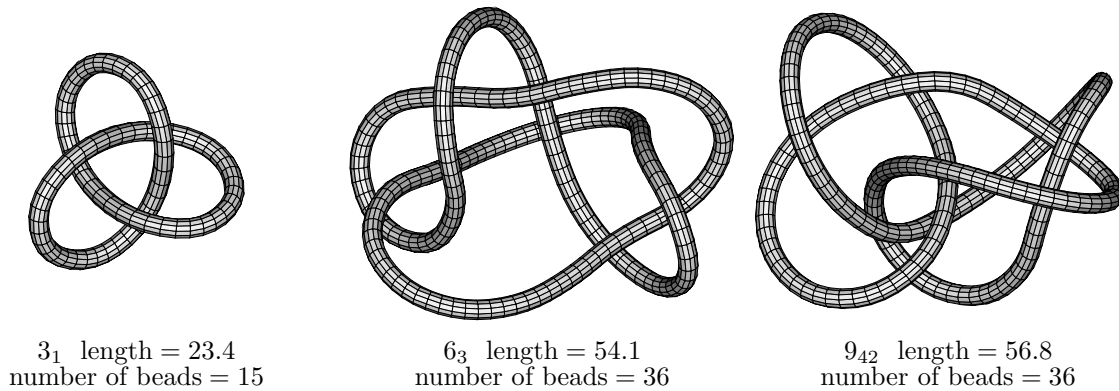
Figure 7.16: Knots relaxed to a natural length using auto-splitting.

stuck. However, as more beads are added from splitting edges, fewer beads become stuck and the splitting eventually stops. Figure 7.16 shows three knots relaxed in this way. Not surprisingly, 6_3 is considerably longer than 3_1 (it takes more "rope" to tie). What is perhaps unexpected is that 9_{42} is approximately the same length as 6_3, even though its crossing number is three greater. This appears to be related to 9_{42} being a non-alternating knot. Non-alternating knots

136

appear to have more conformational freedom than alternating knots [BSS98b]; this is consistent with results found by other groups [KBM$^+$96].

A second application of auto-splitting is to combine it with bead deletion. This technique is useful for knot simplification and recognition. Figure 7.17 shows the monster unknot once again simplified to a pentagonal knot (which must be the unknot). Phases of relaxation with auto-splitting are alternated with phases of bead deletion. It can be seen that before any beads are deleted, the relaxation proceeds slowly as it did in the example shown in Figure 7.5. However, after some beads are deleted the simplification is quick.

We close this section with two less trivial examples of using the splitting and deleting method for the purpose of knot recognition, and an example of where our methods *fail*. Figure 7.18 shows a complicated unknotted "noose" from the paper of Ligocki and Sethian [LS94].[5] A difficult trefoil, also from [LS94], is shown in Figure 7.19. In both cases, the relaxation proceeds quickly to a simple form where the knot type is easily recognized. On a typical work station, the real time taken by the entire procedure is less than a minute in each case. This is much faster than the times reported by Ligocki and Sethian for their method. However, in all fairness, the problem being solved is different. Ligocki and Sethian are interested in maintaining a strict physical basis for their dynamics, whereas here we are simply interested in the problem of knot recognition.

The example of failure is shown in Figure 7.20 which shows one of the unknots that lack n-waves from the paper of Ochiai [Och90]. The specific embedding was constructed for use in the research presented in Huang, Kauffman, and Grzeszczuk [GHK97]. The knot data was graciously provided to this author by Louis Kauffman. Even with the use of bead tricks, this knot has foiled KnotPlot's attempts to untie it. This failure might be due to an intrinsic weakness in the methods presented here, or perhaps it could be that an insufficient amount of computer time was expended on the case. Although they successfully untie the Ochiai unknot, [GHK97] report that the procedure took 292,000 steps of their algorithm and 107 hours of wall clock time. We have not run KnotPlot for more than a few hours on this knot, and after that

---

[5]Knot data provided courtesy of Terry Ligocki.
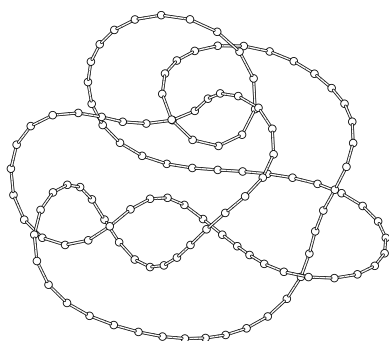
$T = 0, \text{number of beads} = 116$ $\xrightarrow{\text{relax}}$ $T = 33, \text{number of beads} = 116$
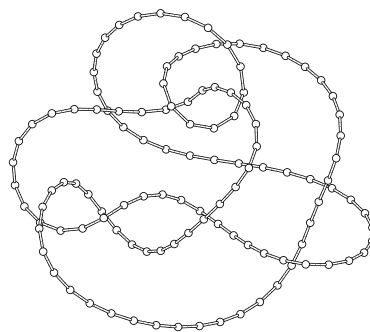
$\searrow$ delete
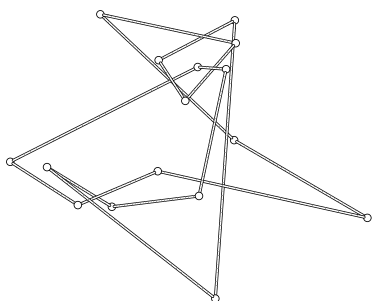
$T = 33, \text{number of beads} = 16$ $\xrightarrow{\text{relax}}$ $T = 66, \text{number of beads} = 24$
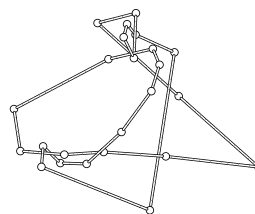
$\searrow$ delete
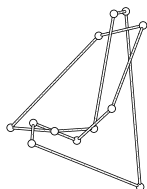
$T = 66, \text{number of beads} = 12$ $\xrightarrow{\text{relax}}$ $T = 99, \text{number of beads} = 16$

$\searrow$ delete

$T = 99, \text{number of beads} = 5$ $\xrightarrow{\text{relax}}$ $T = 132, \text{number of beads} = 5$

Figure 7.17: Relaxation of the "monster" using bead tricks.

138

$T = 0, \text{number of beads} = 160$ $T = 39, \text{number of beads} = 167$

$T = 39, \text{number of beads} = 58$ $T = 78, \text{number of beads} = 98$

$T = 78, \text{number of beads} = 51$ $T = 117, \text{number of beads} = 79$

$T = 117, \text{number of beads} = 25$ $T = 156, \text{number of beads} = 42$

$T = 156, \text{number of beads} = 6$ $T = 195, \text{number of beads} = 6$
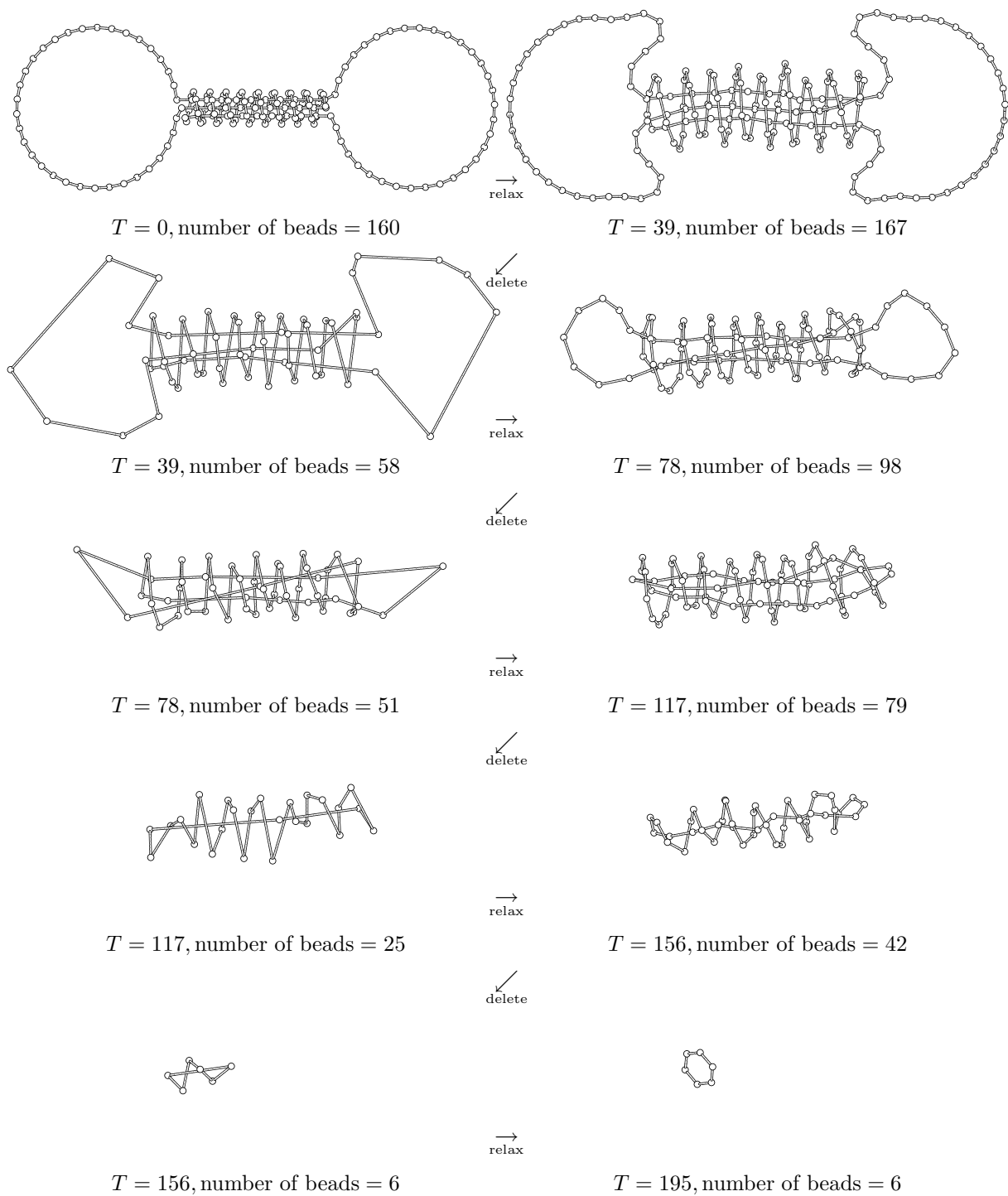
Figure 7.18: Relaxation of Nasty Unknot

$T = 0$, number of beads = 162

$\xrightarrow{\text{relax}}$

$T = 66$, number of beads = 434

$\swarrow_{\text{delete}}$

$T = 66$, number of beads = 143

$\xrightarrow{\text{relax}}$

$T = 132$, number of beads = 344

$\swarrow_{\text{delete}}$

$T = 132$, number of beads = 90

$\xrightarrow{\text{relax}}$

$T = 198$, number of beads = 192

$\swarrow_{\text{delete}}$

$T = 198$, number of beads = 34

$\xrightarrow{\text{relax}}$

$T = 264$, number of beads = 72

$\swarrow_{\text{delete}}$

$T = 264$, number of beads = 6

$\xrightarrow{\text{relax}}$

$T = 330$, number of beads = 14
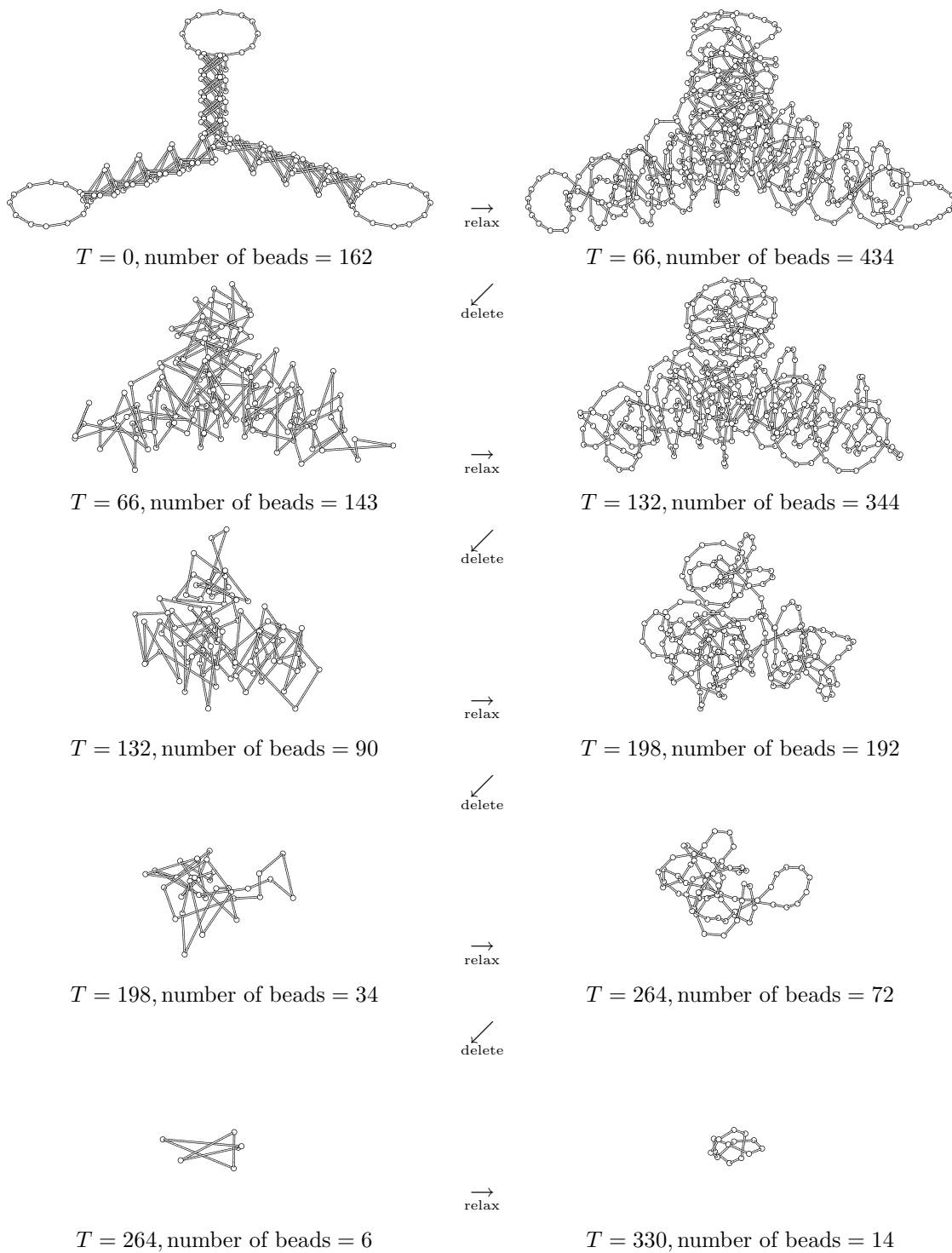
Figure 7.19: Relaxation of Nasty Trefoil

140
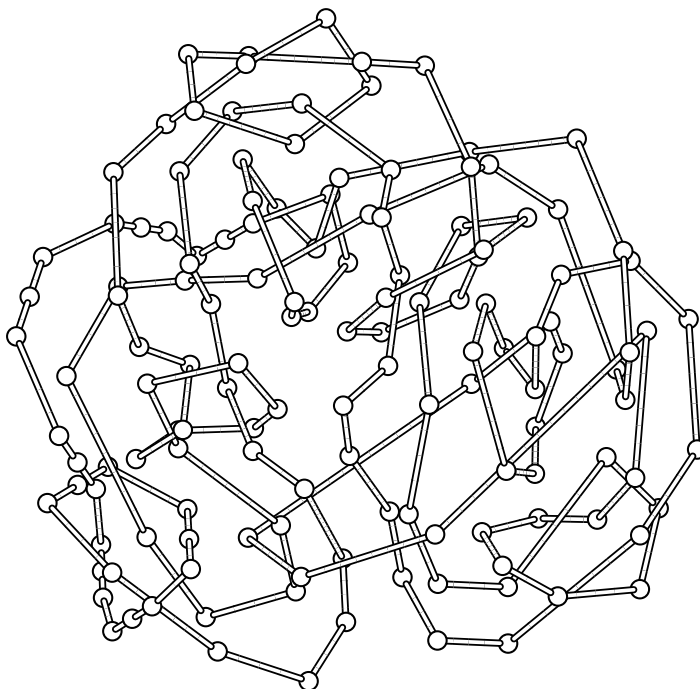
Figure 7.20: Really nasty unknot that foils KnotPlot's dynamics.

time the knot seems to remain in a quasi-stable state, with no net simplification. Of course it is possible that the knot will reach a state where the obstruction to simplification is removed. At that point the knot would quickly simplify. This was the case when *interactive* methods were used for this knot. Using the push/pull rockets, it is easy to untie this knot. One simply "drags" a strand around a part of the knot that is in the way and the knot rapidly simplifies.

Chapter 8

Applications

This chapter presents several applications of topological drawing and refinement to problems in knot theory. Chapter 7 showed the effectiveness of our techniques in the knot recognition problem. Those results, although interesting in their own right, are more significant from a computational viewpoint rather than a mathematical one. In this chapter, we are generally concerned with the use of topological drawing in new research in mathematics.

The first of these research areas is in finding minimal-stick candidates for a large number of knots. These results can be used to constrain theoretical discussions of stick number, for example providing counter-examples. Our methods compare well to other experimental methods [Mei96], but less favourably to theoretical techniques [ABGW97] (when such techniques are possible).

Also discussed in this chapter are relaxations using the symmetric energy [BSS98b] that was defined in Section 3.5.1. To the author's knowledge, these relaxations have not been performed elsewhere. We find that the symmetric energy provides effective simplification of knots from highly tangled states to states approaching the "canonical form". In several cases, the resulting configurations exhibit curious symmetries that, given the physical significance of the symmetric energy, may indicate interesting directions for further research.

The next application shows the effective use of topological refinement to simplify hyperbolic knots with few ideal hyperbolic tetrahedra in their complements. Here we start with knots

in complex configurations, usually given by a long braid-word, and apply *interactive* topological refinement. In one dramatic case, a knot simplifies from a 157 crossing projection down to a 13 crossing projection.

The final application of topological drawing is to the area of mathematical illustration. It is hoped that this thesis itself provides a convincing evidence of the utility of our techniques, as most of the illustrations in this thesis are produced using the methods described here.

8.1 Stick number experiments

Section 3.3.2 introduced the *stick number* problem for knots: what is the minimum number of straight sticks required to form a representative of a given knot-type? Some results are known from theoretical studies [Ran94, ABGW97] and experiments [Mei96]. We have used KnotPlot to extend the range of known upper-limits on the value for the stick number for the entire Rolfsen catalogue, (Appendix C of [Rol76]), a total of 249 proper knots and 130 proper links. While these results do not contain any new values that are *provably* minimal stick representatives, the existence of known upper-limits can be used to constrain theoretical studies of stick numbers. It is conceivable that many of our minimal-stick candidates are in fact minimal-stick representatives, however, we must wait until sufficiently powerful theoretical methods are available before most of them can be proven to be so. It is unlikely that such an advance will be forthcoming, since most knots (as crossing number increases) do not have any special characteristics that allow the analysis to be manageable. Theoretical studies, as in [ABGW97], have focussed on special classes of knots, such as torus knots or their connected sums. These knots are exceptional in many ways, the stick-number of the "average" knot is probably a much more difficult theoretical problem.

8.1.1 Method

Our method is quite straight-forward, and perhaps somewhat "brute-force" in comparison to the manual manipulation technique of Meissen [Mei96]. KnotPlot could easily be used manually

for finding stick numbers since it does allow direct manipulation of a knot, one bead at at time, while ensuring that the knot-type remains unchanged. Whatever tool is used (Meissen used the KED software [Hun96]), finding stick numbers manually is exceedingly tedious.

$T = 0$, number of beads $= 11$ $T = 240$, number of beads $= 11$ $T = 241$, number of beads $= 10$

$T = 242$, number of beads $= 9$ $T = 273$, number of beads $= 9$ $T = 274$, number of beads $= 8$
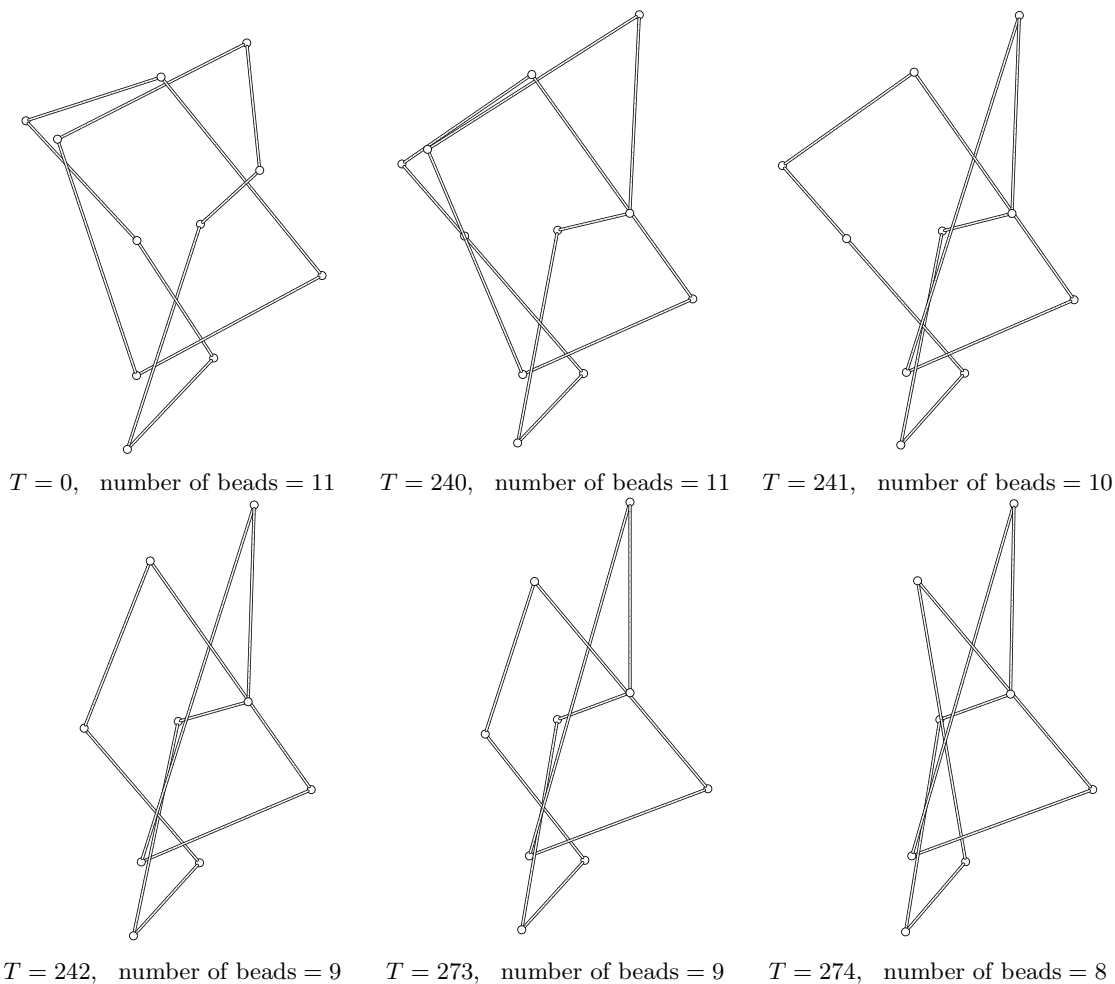
Figure 8.1: Finding a minimal stick representative for 6_2.

The technique used here to find stick-number candidates uses the "thermal" force together with "opportunistic" bead-deletion. Essentially, the following steps are applied in sequence many times:

1. Randomly displace each bead in the knot a small amount using *only* the thermal force. Since this uses the standard KnotPlot dynamics, this preserves knot-type.

2. Attempt to delete as many beads as possible, according to the technique described in Section 7.3.3. This again preserves knot-type.

In our experiments, we have not been greatly concerned with efficiency, instead relying on large amounts of (otherwise unused) computer time. The amount of computation necessary for each candidate depends strongly on several KnotPlot parameters; for example, the magnitude of the thermal force and the number of iteration steps between bead deletion phases, and the specific way the beads are deleted. Although we have attempted to find the best settings for each of the parameters, this matter deserves more careful investigation.
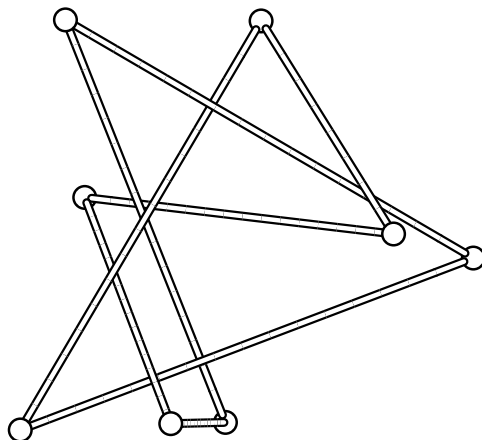
8-stick 6_2, $E_{MD} = 879.2$

Figure 8.2: Minimal stick 6_2 knot at a minimal E_{MD}.

Figure 8.1 shows the procedure for a knot where the stick number is known from theoretical studies. As in previous illustrations, T indicates the number of iteration steps. The $T = 0$ is obtained directly from the knot catalogue representation for 6_2 (which has 82 beads) by deleting as many beads as possible. By $T = 240$, the knot has been "jiggled" to a configuration where one more small displacement will allow one bead to be deleted. After that bead has been deleted ($T = 241$), one more bead can immediately be deleted ($T = 242$). Eventually, at $T = 274$ the knot has reached its minimal number of sticks. The entire process, from $T = 0$ to $T = 274$, takes approximately five seconds on a typical workstation.[1] The example shown is a best case;

---

[1]200 MHz IP22 processor (MIPS R4000/R4400).

145

a typical time would be approximately ten times as long. This is still less than a minute of computation time for these simple knots. In comparison, finding a minimal stick representation manually is likely to take far longer.

After a minimal stick candidate is found, it is worthwhile to take the time to minimize the MD energy. This improves the embedding to make it more "presentable". Also we will see later in this chapter that interesting symmetries are often revealed when the knot is brought to a minimal energy conformation. The result of this procedure is shown for the 6_2 knot in Figure 8.2.

8.1.2 Results

Our results for knots with a crossing number of eight or less, together with a few composite knots and torus knots, are shown in Table 8.1. Meissen [Mei96] has independently found stick numbers for some of these knots (Table 3.2); our numbers agree for all knots examined in both studies. In the case of the seven and eight crossing prime knots, the numbers shown are provisional. It is generally believed that for these knots the result cannot be improved. Our method finds minimal stick representative for both the Granny knot and the Square knot (eight sticks each), however, it fails for both versions of the connected sum of three trefoils. According to the study in [ABGW97], these knots have a stick number of 10.

One interesting observation is that the non-alternating knots (8_{19}, 8_{20}, and 8_{21}) have lower stick-numbers than do most of the other eight-crossing knots. In two cases this number is as small as the stick-number for a six-crossing knot. This result is consistent with the observation made by others [KBM$^+$96] that non-alternating knots have a greater conformational freedom than do alternating knots. Perhaps somewhat surprising, however, is the low value for three of the alternating eight-crossing knots (8_{16}, 8_{17}, and 8_{18}). It is also curious that these are the only alternating eight-crossing knots that have a Conway notation that uses one of the basic polyhedra *other than* the $1^\star$ polyhedron.

Of the torus knots in Table 8.1, our method finds a minimal stick representative for only the first five (3_1, 5_1, 7_1, 8_{19}, and $K_{4,5}$), becoming progressively worse from $K_{5,6}$ to $K_{7,8}$. It

146

knot	sticks		knot	sticks		knot	sticks		knot	sticks
3_1	$6^\star$		7_5	9		8_9	10		8_{20}	$8^\star$
4_1	$7^\star$		7_6	9		8_{10}	10		8_{21}	9
5_1	$8^\star$		7_7	9		8_{11}	10		Granny	$8^\star$
5_2	$8^\star$		8_1	10		8_{12}	10		Square	$8^\star$
6_1	$8^\star$		8_2	10		8_{13}	10		$3_1 \sharp 3_1 \sharp 3_1$	11 (10)
6_2	$8^\star$		8_3	10		8_{14}	10		$3_1 \sharp 3_1 \sharp \overline{3_1}$	11 (10)
6_3	$8^\star$		8_4	10		8_{15}	10		$K_{4,5}$	$10^\star$
7_1	9		8_5	10		8_{16}	9		$K_{5,6}$	13 (12)
7_2	9		8_6	10		8_{17}	9		$K_{6,7}$	16 (14)
7_3	9		8_7	10		8_{18}	9		$K_{7,8}$	19 (16)
7_4	9		8_8	10		8_{19}	$8^\star$		$K_{3,4} \sharp K_{3,4}$	13 (12)

Table 8.1: Provisional stick numbers for knots. A * indicates that the value is known to be best value possible (*i.e.* it is the stick number of the knot). Values in parentheses are stick numbers known from theoretical results.

Sticks	Alternating	Non-alternating	Total
9	4	5	9
10	26	3	29
11	11	0	11
Total	41	8	49

Table 8.2: Provisional stick-number results for the nine crossing knots.

is possible that it is more difficult to find stick numbers for torus knots; the small amount of evidence for this is that of the seven crossing knots, the torus knot 7_1 took by far the most time to find a nine stick representative, requiring more than an hour of computation.

Tables 8.2 and 8.3 shows our provisional results for the nine-crossing and ten-crossing knots, respectively. Again, the (provisional) stick-numbers are considerably lower for the non-alternating knots, which become an increasingly larger fraction of all knots as crossing-number increases. Most of the non-alternating nine-crossing and ten-crossing knots have a stick number more typical of an alternating knot with a crossing-number two less.

Figure 8.3 shows one minimal-stick representative (8_{20}) and three minimal-stick candidates (7_7, 9_{41}, and 10_{140}). Because the diagrams in Figure 8.3 are orthographic projections, it

8-stick 8_{20}

9-stick 7_7

9-stick 9_{41}

10-stick 10_{140}
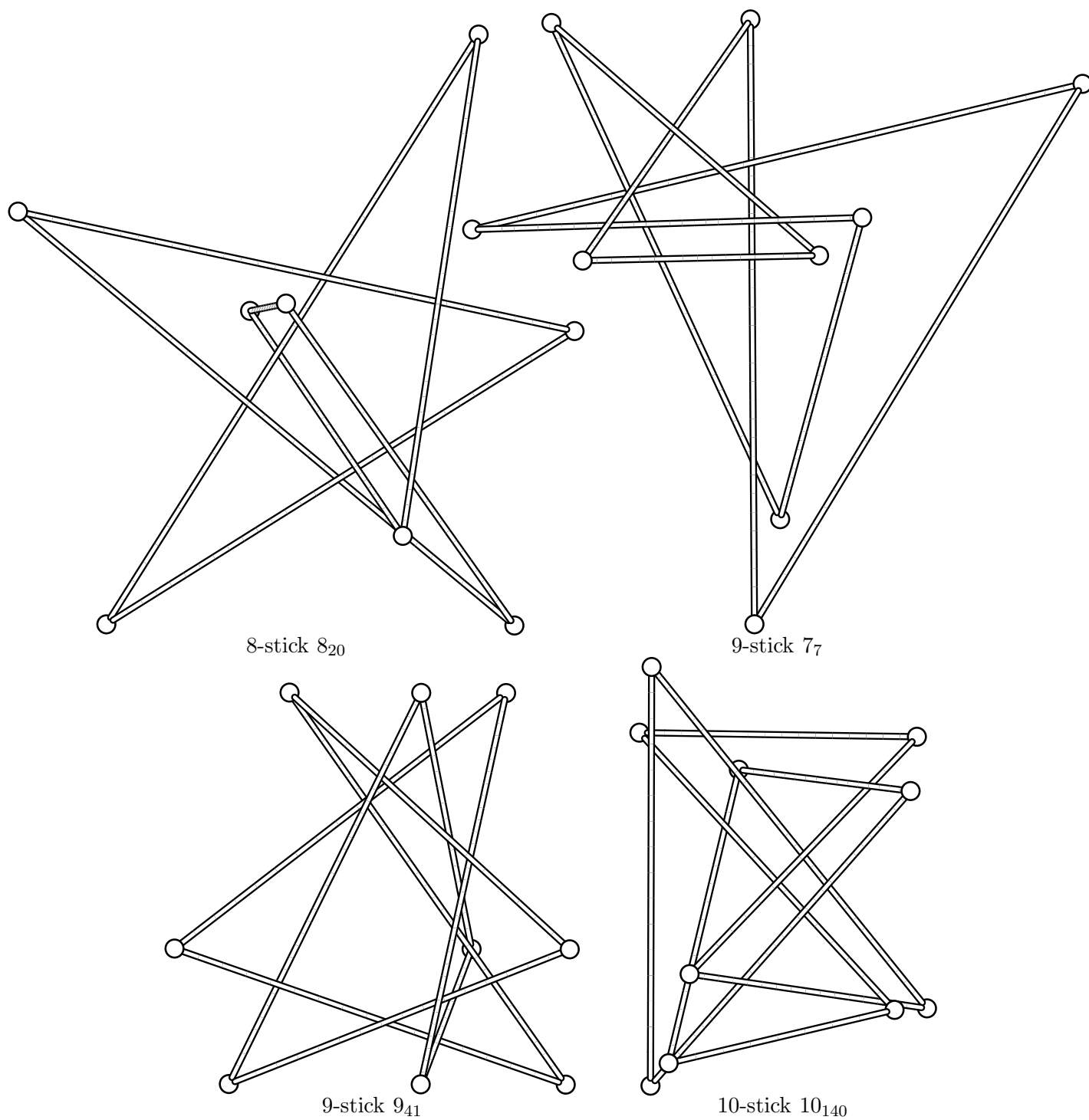
Figure 8.3: Minimal-stick representatives and candidates.

Sticks	Alternating	Non-alternating	Total
10	13	29	42
11	54	13	67
12	50	0	50
13	5	0	5
14	1	0	1
Total	123	42	165

Table 8.3: Provisional stick-number results for the ten crossing knots. The single "problem" knot of 14 sticks is 10_{84}.

is difficult to appreciate the true 3D placement of the beads. Appendix A has stereo-pairs of all knots up to eight crossings, along with several other interesting knots. In those illustrations, it is easier to see that the displacements perpendicular to the plane are often quite large in comparison to distances in the plane.

Several knots in the catalogue exhibit interesting symmetries when brought to a configuration that minimizes the minimum-distance energy E_{MD}. For example, the minimal-stick candidate for 10_{124} shown in Figure 8.4 exhibits a symmetrical projection in three mutually orthogonal projections. Furthermore, the knot in Figure 8.4 has been aligned so that its *principal axes*[2] coincide with the coordinate axes. As shown in the figure, an orthographic projection onto a plane perpendicular to any of these principal axes has bilateral or two-fold rotational symmetry about the principal axis. In Figure 8.4, the z-axis is the principal axis with the largest eigenvalue, and the y-axis has the smallest eigenvalue.

Another set of knots with intriguing symmetries after being brought into a minimum E_{MD} configuration and viewed along the principal axes are the three torus knots 3_1 ($K_{2,3}$), 8_{19} ($K_{3,4}$), and $K_{4,5}$. The stick number of these knots is known from theoretical studies [ABGW97] and our technique finds minimal-stick representatives. These knots are shown in Figure 8.5. A complete catalogue of all knots and links found with these symmetries is included in Appendix A.

---

[2]These are the eigenvectors of the inertia tensor, where the equal masses are located at the bead positions.

149
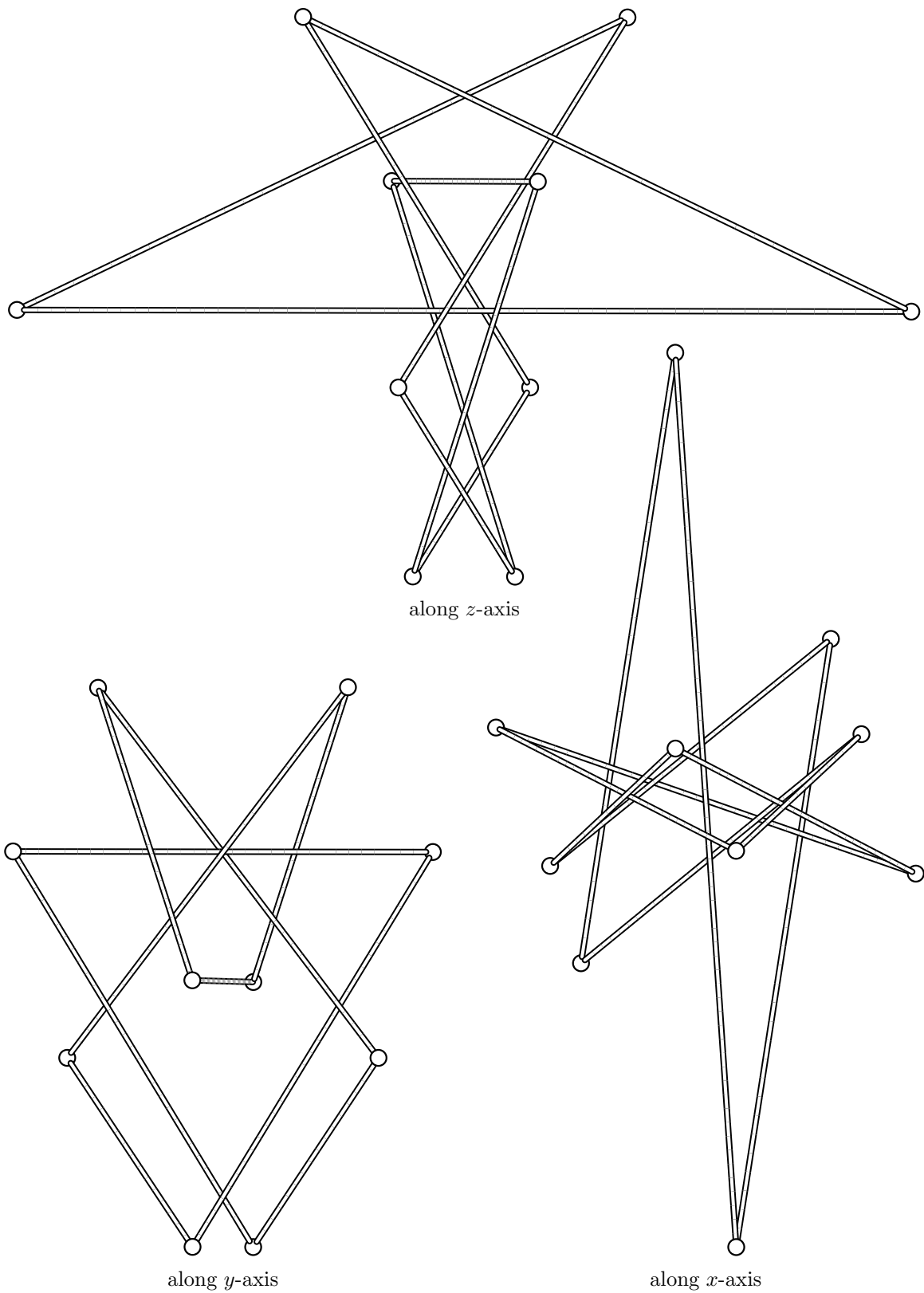
Figure 8.4: Minimal-stick candidate for 10_{124} viewed along its three principal axes.
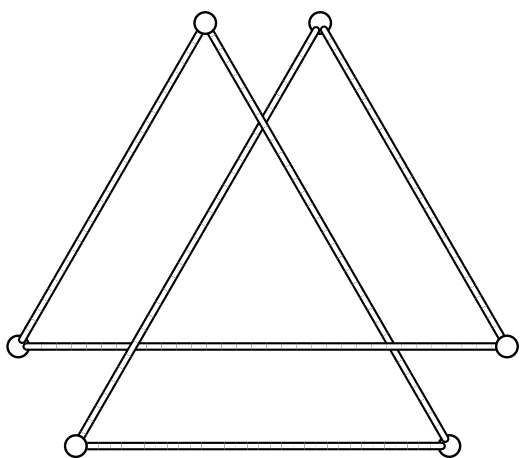
$K_{2,3}$ viewed along principal axis
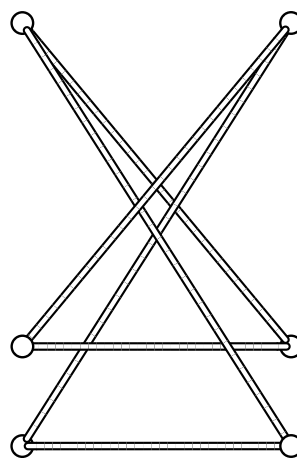
$K_{2,3}$ rotated 90°
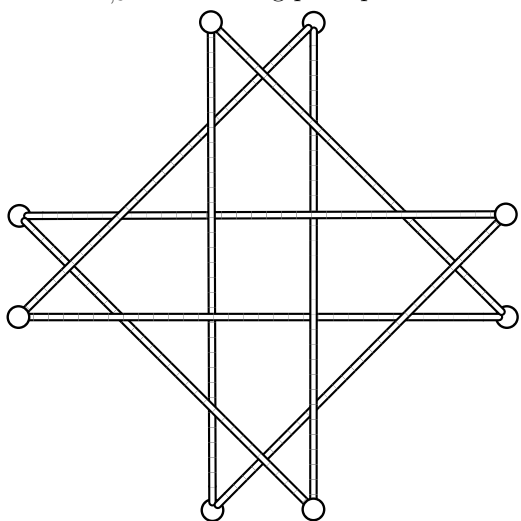
$K_{3,4}$ viewed along principal axis
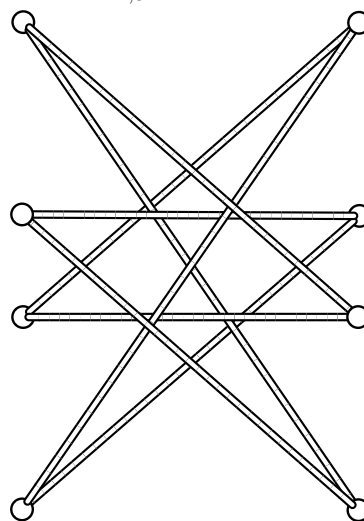
$K_{3,4}$ rotated 90°
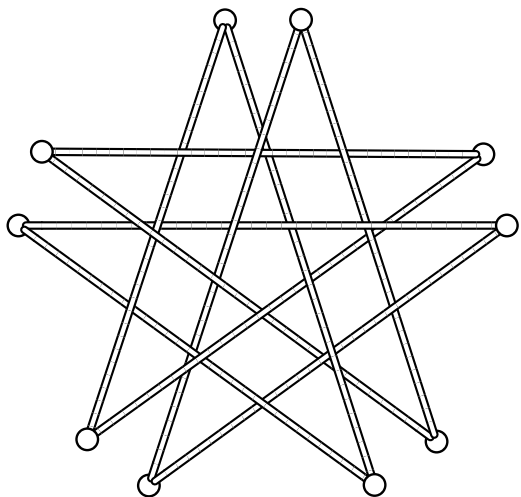
$K_{4,5}$ viewed along principal axis
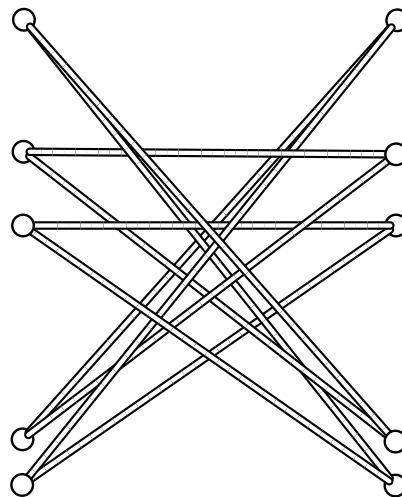
$K_{4,5}$ rotated 90°

Figure 8.5: Three minimal-stick torus knots in minimal E_{MD} configuration.

8.2 Relaxations using the symmetric energy

8.2.1 Method and results

We have performed relaxations using the symmetric energy [BSS98b] as an alternative to Knot-Plot's standard dynamics. The symmetric energy was defined in Section 3.5.1 for the continuous case; we have used a discrete approximation to the gradient of the symmetric energy to implement a new force law in KnotPlot. This new force replaces the "electrical" force of Equation 7.2 as the repulsive component. The "mechanical" force of Equation 7.1 is used in conjunction with the symmetric energy force, with $\beta = 0$ to ensure linear springs.

3_1 length $= 23.4$ 6_3 length $= 57.0$ 9_{42} length $= 65.0$
number of beads $= 21$ number of beads $= 54$ number of beads $= 59$
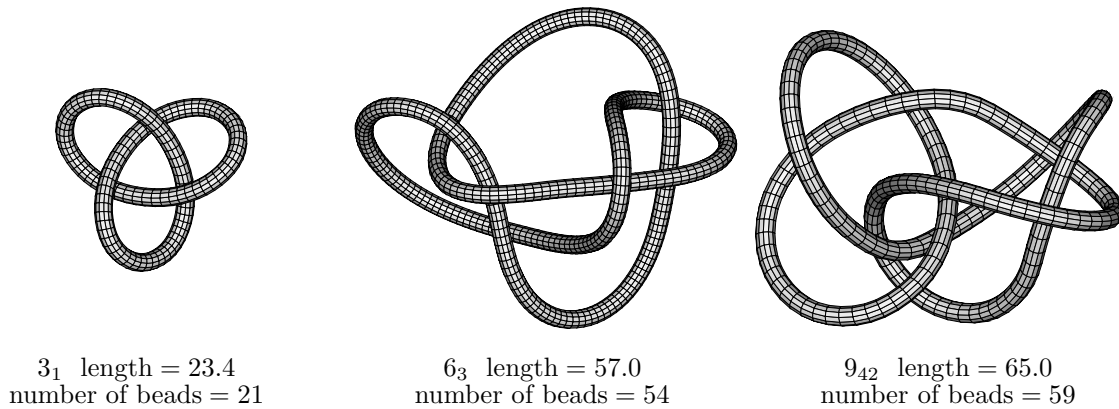
Figure 8.6: Knots relaxed to a natural length using auto-splitting under the symmetric energy.

A similar technique to that previously seen in Section 7.3.3 was used for the production of the catalogue of knots relaxed under the symmetric energy in Appendix A. That technique led to the knot-diagrams in Figure 7.16 which exhibit a "natural-length". For the experiments using the symmetric energy, we again started with minimal-stick candidates for each knot-type (each in a state of minimum E_{MD}) and relaxed while auto-splitting until the knot reached an equilibrium point. Although this method may appear *ad hoc*, it is a reasonable approach given that no obvious technique for specifying initial configurations exists.

Figure 8.6 shows three of these relaxed knots for direct comparison to Figure 7.16, which showed knots relaxed under KnotPlot's standard dynamics. In Figure 8.6 the three knots have been scaled by a constant factor so that the trefoil is the same length as the trefoil in Figure 7.16.

The distance of closest approach is smaller in the symmetric energy case. This is to be expected as the symmetric energy force behaves as $1/r^2$ rather than the $1/r^6$ of the electric force.

Figure 8.7 shows a plot of the symmetric energy $E_S(K)$ versus the average crossing number $\mathcal{ACN}(K)$ for the knots in Appendix A. It is interesting that the data points lie on a nearly straight line. Obviously, this would not be the case for knots in arbitrary configurations, as the average crossing number has a lower bound equal to the crossing number of the knot, and the symmetric energy goes to infinity for knots approaching self-intersection. The only known theoretical relationship between the two quantities is that the symmetric energy bounds the average crossing number, $E_S(K) \geq 4\pi\mathcal{ACN}(K)$, as is shown in [BS97, BSS98b]. In Figure 8.7, this would imply that all data points should lie to the left of the line drawn. The linear relationship shown for the relaxed knots in the plot is therefore somewhat unexpected, however, a similar linear relationship between a different knot energy (thickness energy) and the average crossing number for knots relaxed under that energy have been observed [KBM+96].

The symmetric energy appears to be equally good as KnotPlot's standard dynamics at untangling and simplifying knots, as Figure 8.8 shows. Other relaxations of tangled messes can be seen as MPEG movies at the *Symmetric Energy* World Wide Web site (see Appendix B for the address).

8.2.2   Achiral knots

Two knots in the catalogue relaxed to states that reveal interesting symmetries. The figure-eight knot and the knot 8_3 each relax to conformations that exhibit *rigid geometric achirality*, as shown in Figures 8.9 and 8.10, respectively. In each case a rotation about the indicated axis by $180°$ maps the knot into its mirror image. This was not observed to happen for the other achiral knots that were relaxed (for example 6_3).

It is interesting to view these knots along their three principal axes as we did before for 10_{124} and the three torus knots. Figures 8.11 and 8.12 show the (orthographic) projections along each principal axis. Several new symmetries are now apparent. A rotation of $90°$ about the y-axis, which is the principal axis with the largest moment of inertia, brings the knots into

Symmetric Energy vs. Average Crossing Number

Average Crossing Number
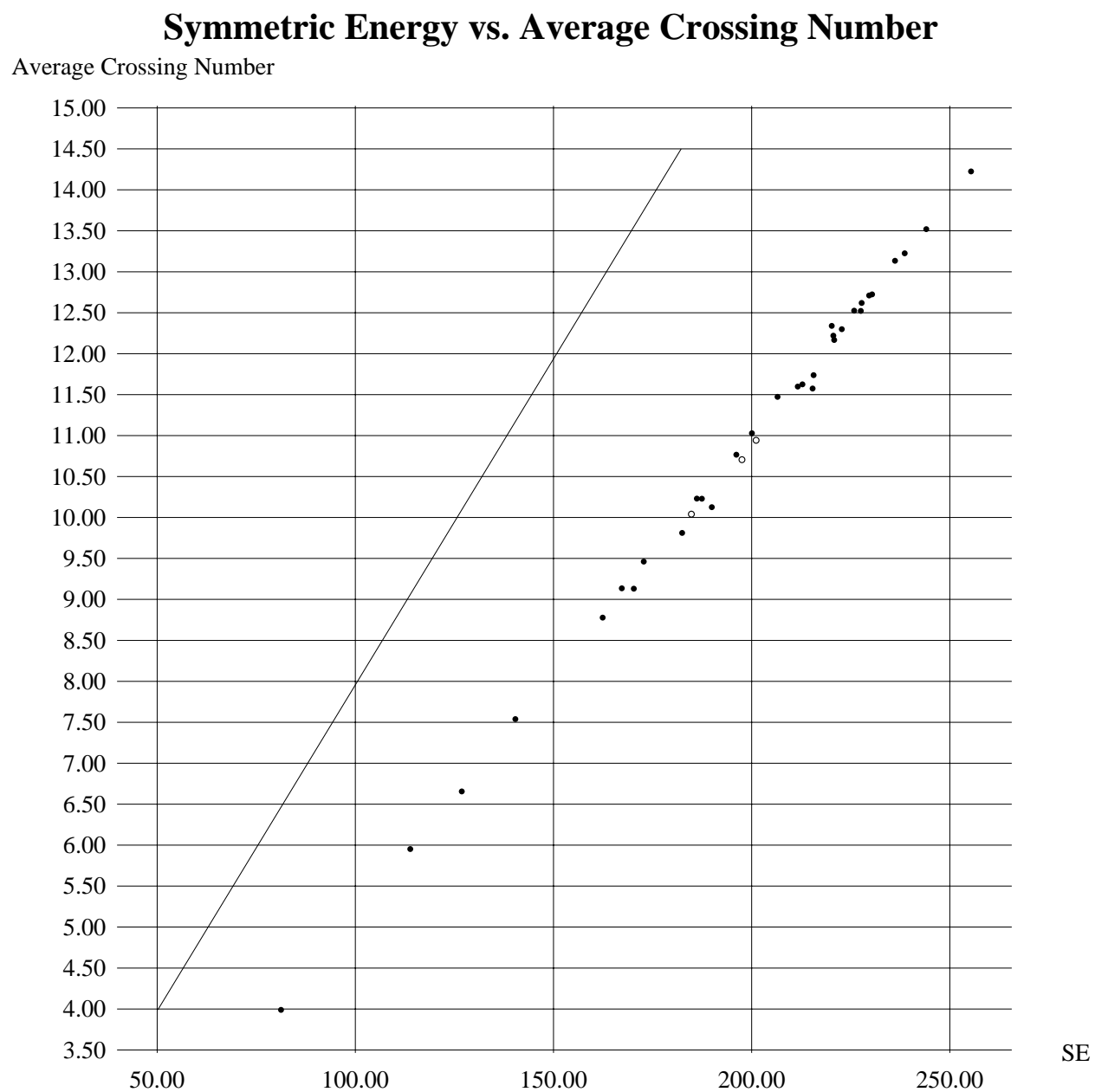
Figure 8.7: Symmetric energy versus average crossing-number of prime knots up to eight-crossings relaxed under symmetric energy. The three non-alternating knots (8_{19}, 8_{20}, and 8_{21}) are indicated by open circles.

$T = 0, E_S = 2597$

$T = 2, E_S = 2407$

$T = 6, E_S = 2183$

$T = 11, E_S = 2005$

$T = 19, E_S = 1814$

$T = 34, E_S = 1592$

$T = 59, E_S = 1392$

$T = 104, E_S = 1196$

$T = 169, E_S = 1002$

$T = 317, E_S = 798$

$T = 798, E_S = 600$

$T = 6918, E_S = 403$
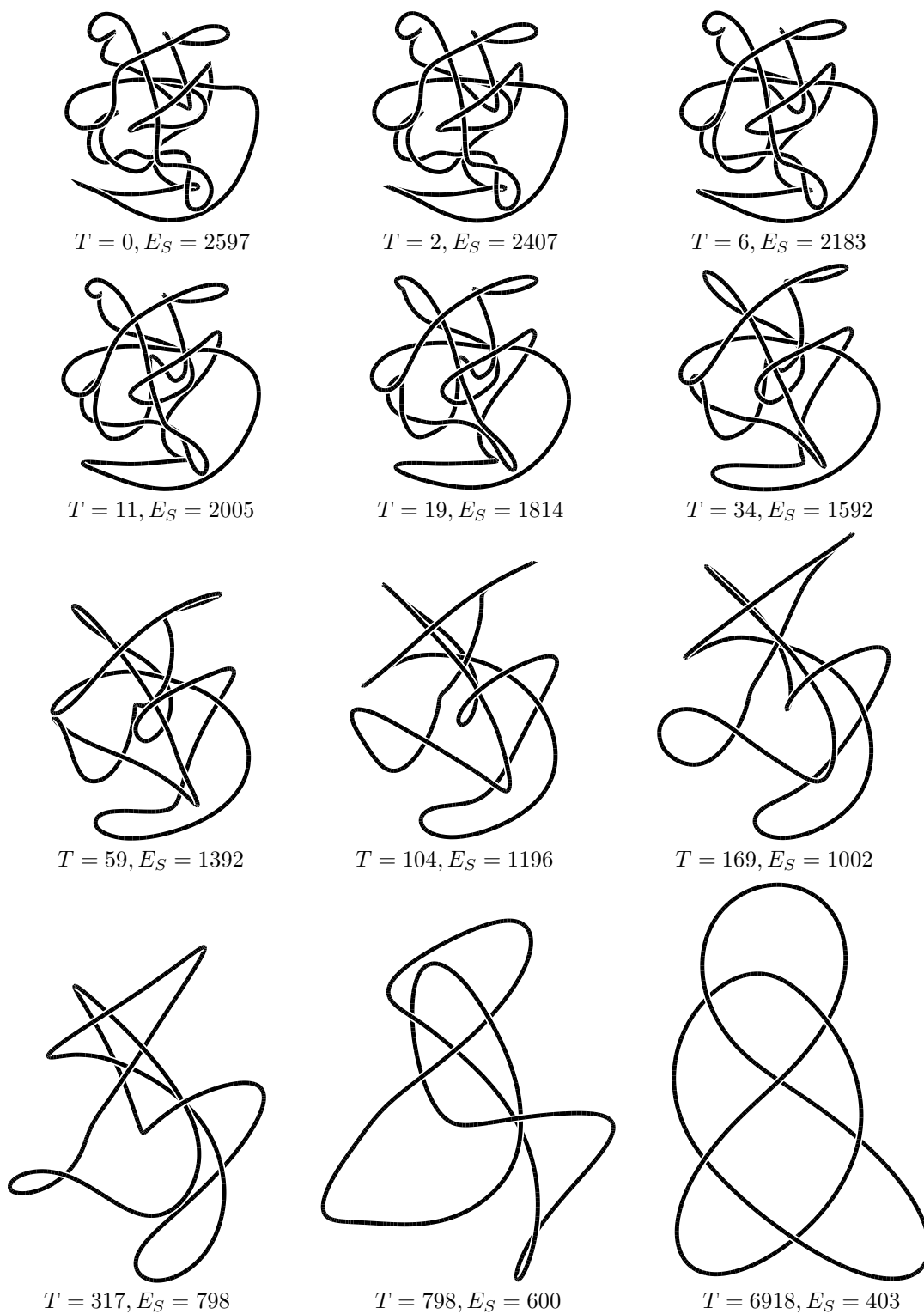
Figure 8.8: Relaxation of the knot 5_2 under the symmetric energy, E_S, shown at approximately equal steps in E_S.
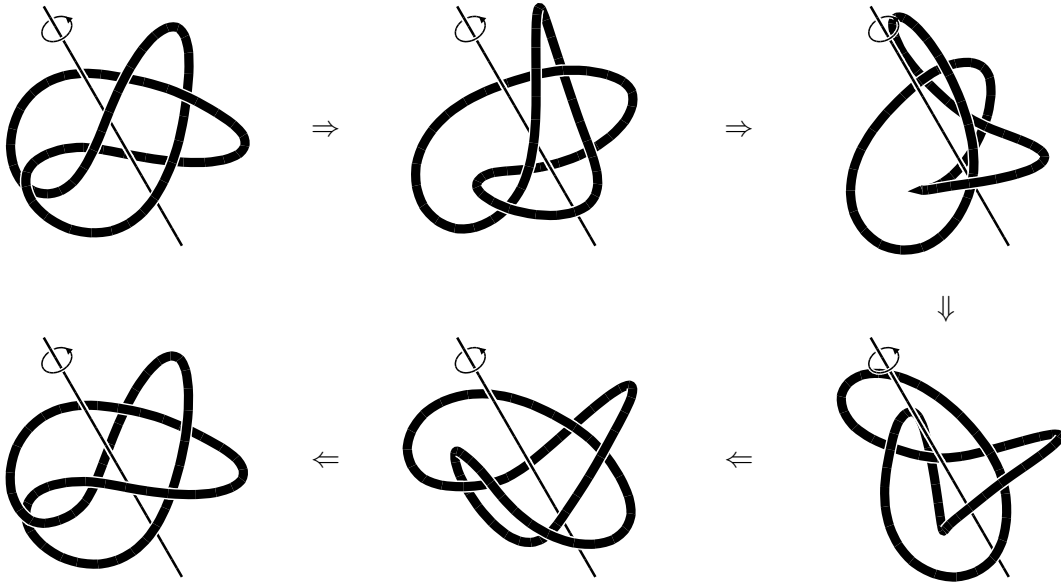
Figure 8.9: Rigid geometric achirality of figure-eight knot, after relaxation under the symmetric energy.
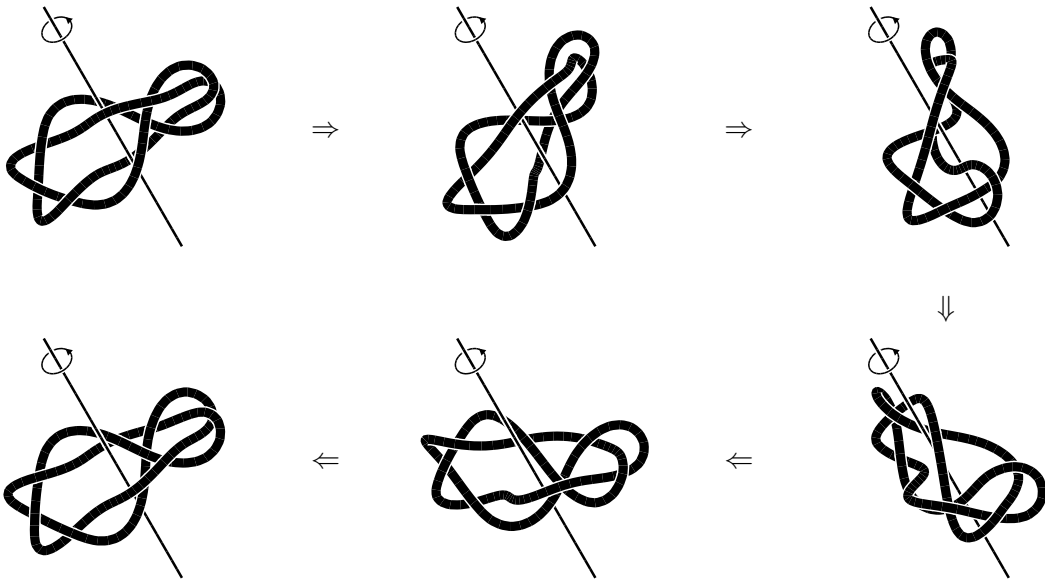
Figure 8.10: Rigid geometric achirality of 8_3 knot, after relaxation under the symmetric energy.
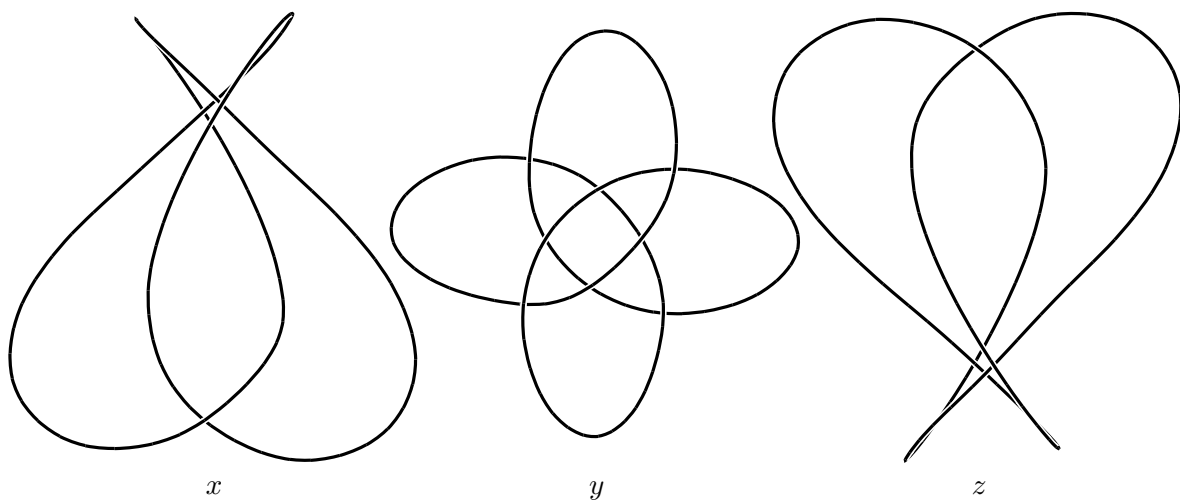
Figure 8.11: Relaxed figure-eight knot viewed along principal axes.
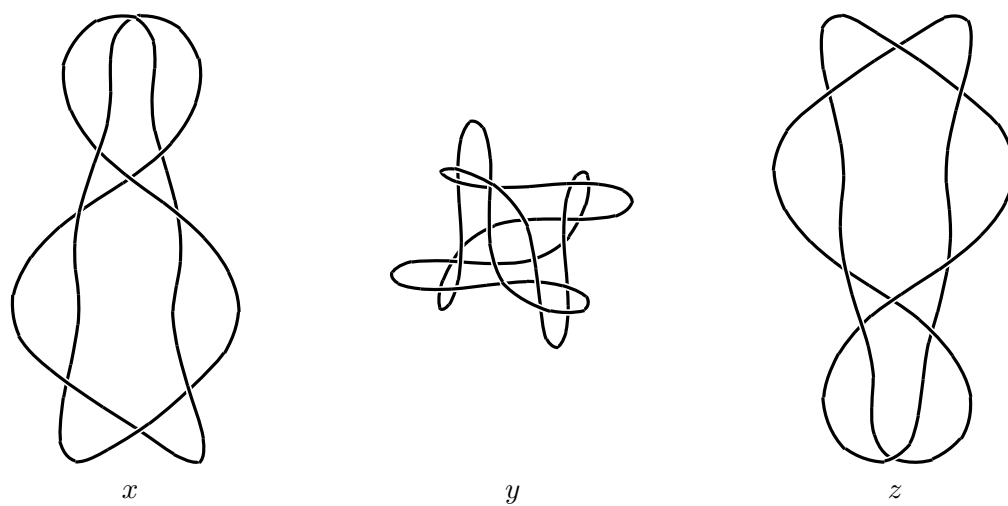
Figure 8.12: Relaxed 8_3 knot viewed along principal axes.

their mirror images. Each of these knots then has *two* different axes about which a rigid rotation maps the knot into its mirror image. The axis of rotation shown in Figures 8.9 and 8.10 is the bisector between the x and z principal axes.

8.2.3 Visualizing the symmetric energy: knots as radiating tubes

The formulation of the symmetric energy due to Buck [BS97] conceives of a knot as a *radiating tube.* In this model the tube is thought of as being a semi-transparent, diffusive surface (something like cellophane) of a small radius r, inside of which (along the core of the tube) lies the knot itself. Such an illumination model corresponds exactly to the computer graphics notion of raytracing, in which light sources are generally not seen, but illuminate surfaces notwithstanding. This correspondence allows us to create a direct visualization of the symmetric energy for any knot conformation. Identical light sources are strung along the core of the tube at roughly uniform intervals. Shadows are not considered partly for the reason of efficiency, as this makes the raytracing much faster (typically there may be several hundred light sources in one scene) and for the more important reason that shadows are a second order effect, contributing a term of order r^2 in the symmetric energy. Since the symmetric energy is defined as the limit of the self-illumination as $r \rightarrow 0$, the shadow term vanishes.

Colour Plate 2 shows a collection of knots visualized in this way. In these images, the knot is drawn with a tube of appreciable radius (since otherwise we wouldn't see it), however, the reader should bear in mind that the illumination model is strictly valid for a thin tube. In fact, the illustrations in Colour Plate 2 shows us not only the symmetric energy, but also the *symmetric energy density* along the tube. Regions of the tube that are bright are those areas where more of the rest of the knot can be seen; *i.e.* the knot is bright near regions of close approach, "extremities" glow dimly. This is especially evident in Colour Plate 2(e), which shows Ligocki and Sethian's "noose" seen in the last chapter (Figure 7.18). The tightly coiled portion glows brightly, as if it were a heating element or a light bulb filament. The multi-coloured pictures seen on the right hand side are links where each component is coloured separately. Colour Plate 2(c) is a homage to Tait. He was probably the first to draw this knot in the form

shown, although in [Tai$\overline{76}$] it wasn't cabled as it is here.

High quality versions of all of these images, and many more besides, are available at the previously mentioned WWW site. A complete exposition on knots as radiating tubes is forthcoming in a paper by Buck, Simon, and the author of this thesis [BSS98b].

8.3 Hyperbolic knot census

Perhaps the most extensive test of *interactive* topological refinement has been to the simplification of complicated knots appearing in Callahan, Dean, and Weeks study of hyperbolic knots that have few ideal tetrahedra in their complement [CDW98]. It would go far beyond the scope of this thesis to explain precisely what this means. *Very briefly*, if a knot is viewed as living in $\mathcal{S}^3$ rather than $\mathcal{R}^3$, then the complement of a tubular neighbourhood of the knot (*i.e.* "fatten" the knot much like the "smooth tube" pictures, and exclude these points) is a compact three-manifold with torus boundary. For almost all knots (torus knots and satellite knots are the exceptions), this three-manifold can be endowed with a hyperbolic metric of constant curvature -1, more-over, there exists a canonical triangulation of this manifold using ideal hyperbolic tetrahedra (ideal because their vertices are at infinity). Callahan, Dean, and Weeks use these canonical triangulations to rank knots according to the number of ideal tetrahedra in their complements. This study is interesting because it looks at knot complexity from a completely different viewpoint, opposite to the way we normally look at knots. In some ways it is more satisfying, because there exist powerful computational tools (Weeks' program SnapPea) that can construct the ideal triangulation directly. This is in contrast to the difficulty in computing the crossing number of a knot, a problem for which there are no known algorithms.

It turns out that the knot with the simplest complement, and therefore the simplest knot from this standpoint, is the figure-eight knot. It is the only knot whose complement consists of only two ideal tetrahedra. Callahan, Dean, and Weeks find that there are exactly 72 knots that have complements with six of fewer ideal tetrahedra. It would be unfair to spoil the enjoyment of reading their paper[3] and seeing the catalogue, so Figure 8.13 shows only the first

---

[3]At the time of writing of this thesis, the paper was still unpublished.

seven knots in order of increasing complexity. In the notation for the knots, the first number
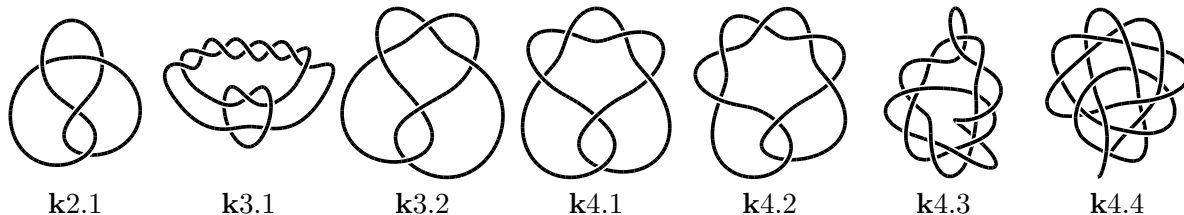
Figure 8.13: The first seven knots in order of increasing complexity of their complement.

denotes the number of ideal tetrahedra in the canonical triangulation. These are the only knots whose complements have fewer than five ideal tetrahedra (there are 22 with five tetrahedra and 43 with six). Many of the knots found by [CDW98] already existed in the standard catalogues; for example, **k**2.1, **k**3.2, **k**4.1, and **k**4.2 are the twist knots 4_1, 5_2, 6_1, and 7_2 from the Rolfsen catalogue [Rol76], respectively. The second most complicated knot, **k**3.1, is the pretzel knot with Conway word $-2, 3, 7$. The remaining two knots have a braid word descriptions as $(\sigma_2\sigma_1)^8\sigma_1^2$ and $(\sigma_2\sigma_4\sigma_1\sigma_3)^3(\sigma_1\sigma_2\sigma_3\sigma_1\sigma_2\sigma_1)^2$. These are a from a class of knots known as *twisted torus knots* [Dea97].

There seems to be little relationship between the complexity of a knot in this viewpoint and the standard notion in terms of crossing number. **k**4.4 is quite complicated as a knot, yet has a relatively simple complement. The knot 6_3 on the other hand, appears as **k**6.43 in the catalogue in [CDW98], requiring six ideal tetrahedra in its complement.

The author of this thesis had the privilege of assisting the authors of the study by constructing knot diagrams for them. About two dozen of the knots found in the study already existed in the Rolfsen catalogue, so constructing diagrams for these was trivial. About five knots were given by a Dowker code description. Producing diagrams for these was also straightforward using the methods of Chapter 6. The remaining knots were twisted torus knots, and generally had rather long descriptions as a braid-word. For these knots an initial embedding was constructed directly from the braid word. Then most of the methods of Chapter 7 were applied in an interactive fashion that would be difficult to duplicate. The interaction involved primarily the use of push/pull rockets at opportune moments, when it looked like a major simplification could

take place if the relaxation were steered in a particular direction. Often a major bead deletion phase would follow that would greatly increase the speed of relaxation.

Overall, this approach worked well at simplifying the complicated knot descriptions. Many of the knots could stand further work, perhaps using newly developed tools. The exercise did reveal many weaknesses in the methods of topological drawing developed in this thesis. In particular, better methods of cutting and splicing (while maintaining knot-type) would make the task somewhat easier. There are a few cases where a dramatic reduction in complexity was possible. Figure 8.14 shows before and after pictures of the knot **k**6.10. During the refinement, this knot is reduced from a 157-crossing projection down to only 13 crossings.[4]

---

[4]In the projection shown, this knot has 15 crossings. A rotation brings it to a projection with only 13 crossings. However, it looks nicer as drawn.
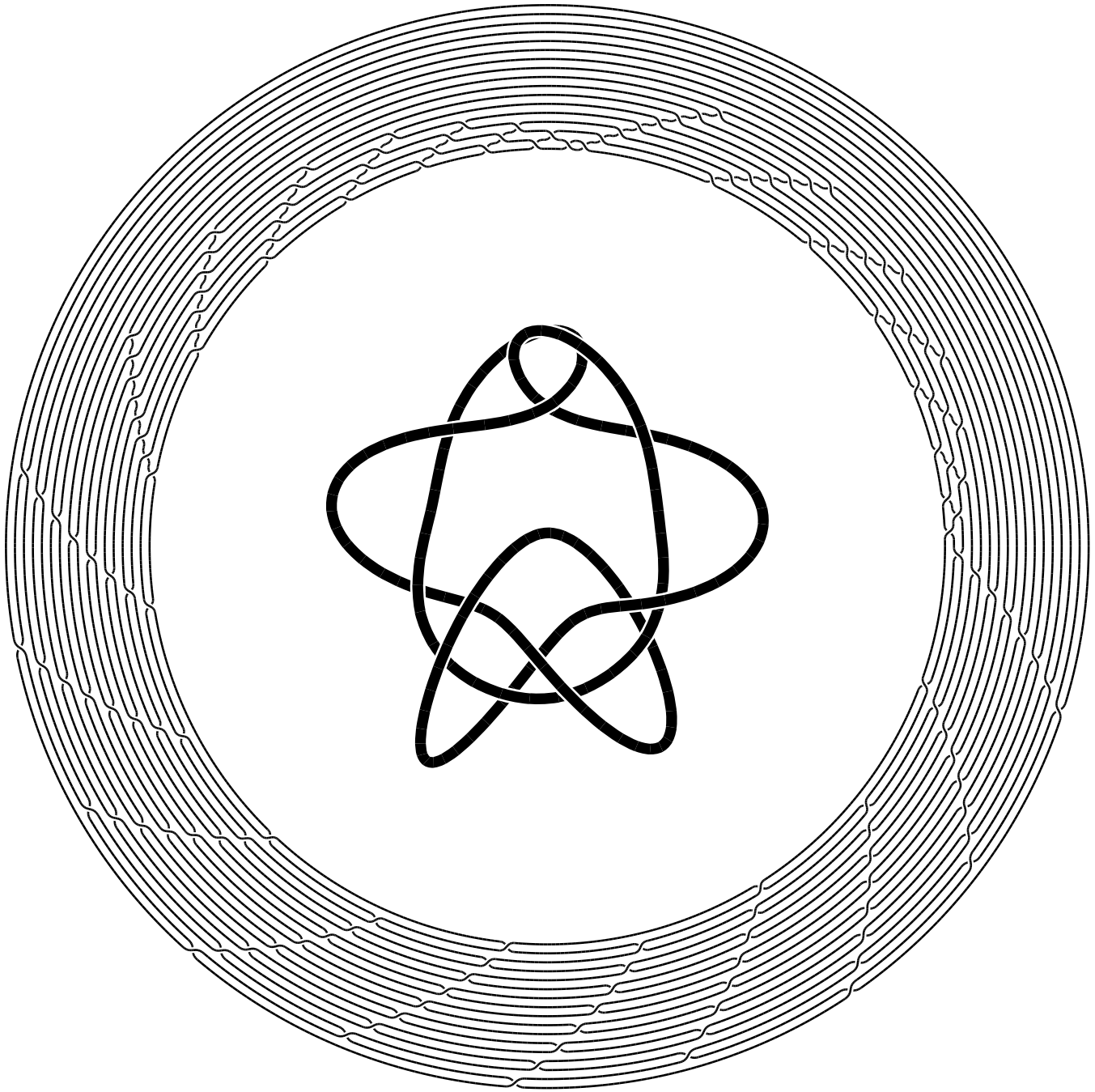
Figure 8.14: Closed braid with word $(\sigma_2^{-1}\sigma_4^{-1}\sigma_6^{-1}\sigma_8^{-1}\sigma_{10}^{-1}\sigma_{12}^{-1}\sigma_{14}^{-1}\sigma_{16}^{-1}\sigma_1^{-1}\sigma_3^{-1}\sigma_5^{-1}\sigma_7^{-1}\sigma_9^{-1}\sigma_{11}^{-1}\sigma_{13}^{-1}\sigma_{15}^{-1})^5\times$
$(\sigma_1\sigma_2\sigma_3\sigma_4\sigma_5\sigma_6\sigma_7\sigma_8\sigma_1\sigma_2\sigma_3\sigma_4\sigma_5\sigma_6\sigma_7\sigma_1\sigma_2\sigma_3\sigma_4\sigma_5\sigma_6\sigma_1\sigma_2\sigma_3\sigma_4\sigma_5\sigma_1\sigma_2\sigma_3\sigma_4\sigma_1\sigma_2\sigma_3\sigma_1\sigma_2\sigma_1)^2$ and the knot after simplification (centre).

Chapter 9

Conclusion and future work

This thesis has demonstrated a reasonably complete approach to topological drawing, within the limited domain of knot theory. We have shown how a wide range of constructions and calculations in knot theory can be simplified using these methods. We have also seen how these techniques can be applied to actual research problems in knot theory. Our results in these areas compare well to others' working in the field.

It is clear that we are still a long way from Roseman's design goal of a general purpose mathematician's drawing tool [Ros92a, Ros92b]. Mathematicians draw and construct an enormous variety of different objects; it is difficult to conceive of a truly complete drawing tool. However, we believe that the research presented here is an important early step in this direction. Perhaps in another decade, topological drawing will be as commonplace as symbolic algebra packages are today.

The remainder of this chapter provides suggestions for future work. Some of these are work in progress, others are more imaginative possibilities.

First of all, it would be interesting to study topological drawing techniques and user interaction with a mathematically sophisticated audience, preferably people with a keen interest in knot theory. Although user studies can be done in the abstract with idealized experimental setups, this author believes that it would be more productive to study actual knot enthusiasts doing real work using topological drawing. Whether such an audience could be found is an open

question.

User studies could also be done with people who don't care much for mathematics, but like to tie knots. In this case the prospective user-base is nearly everybody. Even if people don't like tying knots, they are usually good at tying at least *one* knot. The skills involved in doing these every day tasks (such as tying one's shoes) are incredible. Can we learn to implement something along these lines on a computer?

As for the computational aspects of this work, there are many enhancements that could be made to improve the efficiency. In this thesis we have been more concerned with exploring possibilities rather than maximizing performance. When this work began, there was no anticipation that large-scale projects such as finding minimal stick candidates for hundreds of knots would take place. More sophisticated techniques for finding candidates could be developed and tested against known results from theoretical studies [ABGW97]. A good metric of success would be if the system could rapidly find the theoretical stick numbers. This would be evidence that in cases where nothing is known from theory, the stick number results might provide suggestions for theorems.

Mathematically, there are many avenues for exciting research still open. The four-dimensional aspects of KnotPlot are relatively undeveloped. Construction of arbitrary 4D knots from movies is an easy extension. This combined with the relaxation of such knots in $\mathcal{R}^4$ should yield interesting insights into the 4D world. As for 3D topological applications, an interface between KnotPlot and SnapPea [Wee90] will be developed. Users of both programs can benefit from this connection, KnotPlot users for the sophisticated calculation engine of SnapPea, and SnapPea users for high-quality "input" and "output".

In physical knot theory, relaxations under the symmetric energy deserve more study. Are the curious symmetries found for various knots shown in Chapter 8 unique to the symmetric energy, or are they more universal, perhaps applying to any scale-invariant knot energy? Other knot energies, not discussed in the thesis, are also of interest. In particular, the thickness energy [BS97] could be studied, perhaps exploiting the hardware z-buffer on computer graphics machines for real-time visualization of intersection curves for thick knots. This leads to the study

164

of real (*i.e.* rope, DNA, hair, extension cords) knots as physical objects. As Kauffman [Kau91] correctly points out, a real knot is an extremely complicated physical system.

Of course all of the above can be extended to other topological objects other than knots. Knots have been a fruitful environment for these excursions into topological drawing. From this author's perspective, we have only begun to study knots. Knots are primal shapes; they are everywhere. They capture essential qualities of how objects are embedded in space.
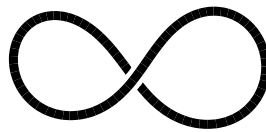
Bibliography

[AB27]   J. W. Alexander and G. B. Briggs. On types of knotted curves. *Ann. of Math. (2)*, 28:562–586, 1927.

[AB90]   William P. Armstrong and Robert P. Burton. Perspective and stereo for projection from and display of four dimensions. In *SPIE Volume 1256: Stereoscopic Displays and Applications*, pages 54–61. SPIE, 1990.

[ABGW97] Colin C. Adams, Bevin M. Brennan, Deborah L. Greilsheimer, and Alexander K. Woo. Stick numbers and composition of knots and links. *Journal of Knot Theory and Its Ramifications*, 6(2):149–161, 1997.

[AC59]   J. J. Andrews and M. L. Curtis. Knotted 2-spheres in the 4-sphere. *Annals of Mathematics*, 70(3):565–571, November 1959.

[Ada94]   Colin Adams. *The Knot Book*. W. H. Freeman, New York, 1994.

[AH89]   Kenneth I. Appel and Wolfgang Haken. *Every planar map is 4 colorable*, Volume 98 of *Contemporary mathematics*. American Mathematical Society, Providence, R.I., 1989.

[Ale23]   J. W. Alexander. A lemma on systems of knotted curves. *Proc. Nat. Acad. Sci. USA*, 9:93–95, 1923.

[Ale28]   J. W. Alexander. Topological invariants of knots and links. *Transactions of the American Mathematical Society*, 30:275–306, 1928.

[Ali98]   Alias. Alias|Wavefront home page, 1998. World Wide Web site http://www.alias.com

[Alt86]   Simon L. Altmann. *Rotations, Quaternions, and Double Groups*. Clarendon Press, Oxford, 1986.

[APR89]   R. P. Anstee, J. H. Przytycki, and D. Rolfsen. Knot polynomials and generalized mutation. *Topology and its Applications*, 32:237–249, 1989.

[Art25]   E. Artin. Theorie der Zöpfe. *Abh. Math. Sem. Univ. Hamburg*, 4:47–72, 1925.

[Art47]   E. Artin. Theory of braids. *Ann. of Math. (2)*, 48:101–126, 1947.

[Ash44]   Clifford W. Ashley. *The Ashley Book of Knots.* Doubleday, New York, Corrected edition, 1944.

[Aut97]   AutoCad. Autocad reference manual, 1997.

[Bac81]   Janet Backhouse. *The Lindisfarne Gospels.* Cornell University Press, Ithaca, New York, 1981.

[Bai73]   George Bain. *Celtic Art — The Methods of Construction.* Dover, New York, 1973. Unabridged republication of the work originally published by William MacLellan & Co., Ltd., Glascow, in 1951.

[Bai86]   Iain Bain. *Celtic Knotwork.* Constable, London, 1986.

[Ban86]   Thomas F. Banchoff. Visualizing two-dimensional phenomena in four-dimensional space: a computer graphics approach. In Edward Wegman and Douglas DePriest, editors, *Statistical Image Processing and Graphics*, pages 187–202. Marcel Dekker, Inc., New York, 1986.

[Ban90]   Thomas F. Banchoff. *Beyond the Third Dimension — Geometry, Computer Graphics, and Higher Dimensions.* W. H. Freeman and Company, New York, 1990.

[Ban92]   David Banks. Interactive manipulation and display of two-dimensional surfaces in four-dimensional space. In *Proceedings 1992 Symposium on Interactive 3D Graphics*, pages 197–207, Cambridge, Massachusetts, 1992.

[Ban93]   David Banks. *Interacting with Surfaces in Four Dimensions Using Computer Graphics.* PhD thesis, Department of Computer Science, University of North Carolina at Chapel Hill, March 1993. Also available as technical report TR93-011.

[Bar94]   A. H. Barr. Global and local deformations of solid primitives. In *Computer Graphics Proceedings (SIGGRAPH 84)*, Annual Conference Series, pages 21–30. ACM SIGGRAPH, 1994.

[BC92]   Thomas F. Banchoff and Davide P. Cervone. Illustrating "Beyond the Third Dimension". *Leonardo*, 25(3/4):273–280, 1992.

[BF88]   Clifford M. Beshers and Steven K. Feiner. Real-time 4D animation on a 3D graphics workstation. In *Proceedings of Graphics Interface '88*, pages 1–7, June 1988.

[BHJS94] M. G. V. Bogle, J. E. Hearst, V. F. R. Jones, and L. Stoilov. Lissajous knots. *Journal of Knot Theory and Its Ramifications*, 3(2):121–140, 1994.

[Bir74]   Joan S. Birman. *Braids, Links and Mapping Class Groups.* Number 82 in Annals of Mathematical Studies. Princeton University Press, Princeton, 1974.

[Bir93]   Joan S. Birman. Recent developments in braid and link theory. *Mathematical Intelligencer*, 13:52–60, 1993.

[BO93]   G. Buck and J. Orloff. Computing canonical conformations for knots. *Topology and its Applications*, 51:247–253, 1993.

[BO95]   G. Buck and J. Orloff. A simple energy function for knots. *Topology and its Applications*, 61:205–214, 1995.

[Bot94]   David Botta. 4D. *Centre for Image and Sound Research (CI★SR) Newsletter*, April 1994.

[BR91]   Mark Bloomenthal and Richard Riesenfeld. Approximation of sweep surfaces by tensor product NURBS. pages 132–143, 1991.

[Bra92]   Kenneth A. Brakke. The surface evolver. *Experimental Mathematics*, 1:141–165, 1992.

[Bri78a]   David W. Brisson, editor. *Hypergraphics: Visualizing Complex Relationships in Art, Science and Technology*, Volume 24 of *AAAS Selected Symposium*. Westview Press, Boulder, Colorado, 1978.

[Bri78b]   David W. Brisson. Visual comprehension of n-dimensions. In Brisson [Bri78a], Chapter 6, pages 109–145.

[BS78]   Thomas F. Banchoff and Charles M. Strauss. Real-time computer graphics analysis of figures in four-space. In Brisson [Bri78a], Chapter 8, pages 159–176.

[BS93]   Gregory Buck and Jonathan Simon. Knots as dynamical systems. *Topology and its Applications*, 51:229–246, 1993.

[BS97]   Gregory Buck and Jonathan Simon. Energy and length of knots. In Shin'ichi Suzuki, editor, *Lectures at Knots96*, pages 219–234. MSJ International Conference on Knots, World Scientific, 1997.

[BSS98a]   Gregory Buck, Jonathan Simon, and Robert G. Scharein. Physical knot theory. *Scientific American*, 1998. (To appear).

[BSS98b]   Gregory Buck, Jonathan Simon, and Robert G. Scharein. Physical knots: radiating tubes, rope-length, and crossing number. (Unpublished, submitted to *Nature*), 1998.

[BZ85]   Gerhard Burde and Heiner Zieschang. *Knots*. W. De Gruyter, Berlin, New York, 1985.

[CDW98]   Patrick J. Callahan, John C. Dean, and Jeffrey R. Weeks. The simplest hyperbolic knots. (preliminary version 0.91), 1998.

[CG83]   J. H. Conway and C. McA. Gordon. Knots and links in spatial graphs. *Journal of Graph Theory*, 7:445–453, 1983.

[CMS88]   Michael Chen, S. Joy Mountford, and Abigail Sellen. A study in interactive 3-D rotation using 2-D control devices. In *Computer Graphics Proceedings (SIGGRAPH 88)*, Annual Conference Series, pages 121–129. ACM SIGGRAPH, 1988.

[Con70]   J. H. Conway. An enumeration of knots and links, and some of their algebraic properties. In John Leech, editor, *Computational Problems in Abstract Algebra*, pages 329–358, 1970.

[CON85]   Norishige Chiba, Kazunori Onoguchi, and Takao Nishizeki. Drawing plane graphs nicely. *Acta Informatica*, 22:187–201, 1985.

[Con92]   Patrick Conty. The geometry of the labyrinth. *Parabola*, 17(2):4–14, Summer 1992.

[Coq90]   Sabine Coquillart. Extended free-form deformation: A sculpturing tool for 3D geometric modeling. In *Computer Graphics Proceedings (SIGGRAPH 90)*, Annual Conference Series, pages 187–195. ACM SIGGRAPH, 1990.

[Cow74]   Thaddeus M. Cowan. The theory of braids and the analysis of impossible figures. *Journal of Mathematical Psychology*, 11:190–212, 1974.

[Cow77]   Thaddeus M. Cowan. Supplementary report: Braids, side segments, and impossible figures. *Journal of Mathematical Psychology*, 16:254–260, 1977.

[Cox92]   Donna J. Cox. Caricature, readymades and metamorphosis: Visual mathematics in the context of art. *Leonardo*, 25(3/4):295–302, May 1992.

[Dea97]   John Dean. Hyperbolic knots. In Jeff Weeks, editor, *Workshop on Computational Methods in Three Dimensional Topology*, Berkeley, California, March 1997. Mathematical Sciences Research Center.

[DG96]   Douglas DeCarlo and Jean Gallier. Topological evolution of surfaces. In Wayne A. Davis and Richard Bartels, editors, *Graphics Interface 1996*, pages 194–203, 22–24 May 1996.

[DH96]   Ron Davidson and David Harel. Drawing graphs nicely using simulated annealing. *ACM Transactions on Graphics*, 15(4):301–331, October 1996.

[DH97]   Oliver T. Dasbach and Stefan Hougardy. Does the Jones polynomial detect unknottedness? *Experimental Mathematics*, 6(1):51–56, 1997.

[DNT90]   David P. Dobkin, Stephen C. North, and Nathaniel J. Thurston. A viewer for mathematical structures and surfaces in 3D. *Computer Graphics*, 24(2), March 1990.

[DNT91]   David Dobkin, Stephen North, and Nathaniel Thurston. *Salem User's Guide*. Princeton University, 1991.

[DPS]   Y. Diao, N. Pippenger, and D. W. Sumners. On random knots in R^3. Preprint.

[DT83]   C. H. Dowker and Morwen B. Thistlethwaite. Classification of knot projections. *Topology and its Applications*, 16:19–31, 1983.

[ES87]   C. Ernst and D. W. Sumners. The growth of the number of prime knots. *Math. Proc. Camb. Phil. Soc.*, 102:303–315, 1987.

[ES90]   C. Ernst and D. W. Sumners. A calculus for rational tangles: applications to DNA recombination. *Math. Proc. Camb. Phil. Soc.*, 108:489–515, 1990.

[FC92]   George K. Francis and Brent Collins. On knot-spanning surfaces: An illustrated essay on topological art. *Leonardo*, 25(3/4):313–320, May 1992.

[Fen89]   Roger Fenn. Fibring the complement of the Fenn-Rolfsen link. *Publicasions Matemàtiques*, 33:383–390, 1989.

[FH91]   Michael H. Freedman and Zheng-Xu He. Divergence-free fields: Energy and asymptotic crossing number. *Annals of Mathematics*, 134:189–229, 1991.

[FH92]   Michael H. Freedman and Zheng-Xu He. Research announcement on the "energy" of knots. In Moffatt [Mof92b], pages 219–222.

[FHH$^+$93] M. Formann, T. Hagerup, J. Haralambides, M. Kaufmann, F. T. Leighton, A. Symvonis, E. Welzl, and G. Woeginger. Drawing graphs in the plane with high resolution. *SIAM J. Comput.*, 22(5):1035–1052, October 1993.

[FHW94]   Michael H. Freedman, Zheng-Xu He, and Zhenghan Wang. Möbius energy of knots and unknots. *Annals of Mathematics*, 139:1–50, 1994.

[For62]   M. K. Fort Jr., editor. *Topology of 3-Manifolds and related topics*. Prentice-Hall, Inc., 1962.

[For97]   David R. Forsey. The Dragon Wing, 1997. World Wide Web site http://www.cs.ubc.ca/nest/imager/contributions/forsey/dragon/top.html

[Fox62]   R. H. Fox. A quick trip through knot theory. In Fort Jr. [For62], Chapter 3, pages 120–167.

[FR86]   Roger Fenn and Dale Rolfsen. Spheres may link homotopically in 4-space. *J. London Math. Soc.*, 34:177–184, 1986.

[Fra87]   George K. Francis. *A Topological Picturebook*. Springer-Verlag, 1987.

[FRC92]   Helaman Ferguson, Alyn Rockwood, and Jordan Cox. Topological design of sculptured surfaces. In *Computer Graphics Proceedings (SIGGRAPH 92)*, Annual Conference Series, pages 149–156. ACM SIGGRAPH, 1992.

[Fuk88]   Shinji Fukuhara. Energy of a knot. In Y. Matsumoto, T. Mizutani, and S. Morita, editors, *A Fête of Topology*, pages 443–451. Academic Press, Inc., 1988.

[FW85]   John Franks and R. F. Williams. Entropy and knots. *Transactions of the American Mathematical Society*, 291(1):241–253, September 1985.

[FYH+85]   P. Freyd, D. Yetter, J. Hoste, W. B. R. Lickorish, K. Millett, and A. Ocneanu. A new polynomial invariant of knots and links. *Bulletin of the American Mathematical Society*, 12(2):239–246, April 1985.

[Gau$\overline{33}$]   K. F. Gauss. Zur mathematischen Theorie der electrodynamischen. In *Werke Königl. Gesell. Wiss. Göttingen 1877*, Volume 5, page 605. Georg Olms Verlag, Hildesheim, 1833.

[GH52]   Raoul Graumont and John Hensel. *Encyclopedia of Knots and Fancy Rope Work*. Cornell Maritime Press, Centreville, Maryland, fourth edition, 1952.

[GHK97]   Robert P. Grzeszczuk, Milana Huang, and Louis H. Kauffman. Physically-based stochastic simplification of mathematical knots. *IEEE Transactions on Visualization and Computer Graphics*, 3(3):262–272, July/September 1997.

[GHS97]   Robert W. Ghrist, Philip J. Holmes, and Michael C. Sullivan. *Knots and Links in Three-Dimensional Flows*, Volume 1654 of *Lecture Notes in Mathematics*. Springer-Verlag, Berlin, 1997.

[Gib79]   James J. Gibson. *The Ecological Approach to Visual Perception*. Houghton-Mifflin, Boston, 1979.

[Gib94]   Rick Gibson. Computer generated anaglyphs. *Centre for Image and Sound Research (CI⋆SR) Newsletter*, April 1994.

[GM91]   Charlie Gunn and Delle Maxwell. Not knot, 1991. Computer animated film published by Jones and Bartlett.

[GMW97]   Baining Guo, Jai Menon, and Brian Willette. Surface reconstruction using alpha shapes. *Computer Graphics Forum*, 16(4):177–190, 1997.

[Gun93]   Charlie Gunn. Discrete groups and visualization of three-dimensional manifolds. In *Computer Graphics Proceedings (SIGGRAPH 93)*, Annual Conference Series, pages 255–262. ACM SIGGRAPH, 1993.

[Hae62]   A. Haefliger. Knotted $(4k - 1)$-spheres in $6k$-space. *Math. Ann.*, 75(1):452–466, 1962.

[Hak61]   W. Haken. Theorie der Normalflächen. *Acta Math.*, 105:245–375, 1961.

[Han89]   Vagn Lundsgaard Hansen. *Braids and Coverings*, Volume 18 of *London Mathematical Society Student Texts*. Cambridge University Press, Cambridge, 1989.

[Han92]   Andrew J. Hanson. The rolling ball. In David Kirk, editor, *Graphics Gems III*, pages 51–60. Academic Press, San Diego, 1992.

[HBK$^+$94]   Roland Haynes, Tom Berryhill, Jeff Koftinoff, Rob Scharein, and Malcolm Warwick. Multi-media jazz improvisation, November 1994. Centre for Image and Sound Research (CI⋆SR), Vancouver B.C.

[HC93]   Andrew J. Hanson and Robert A. Cross. Interactive visualization methods for four dimensions. In *Proceedings of the IEEE Visualization '93 Conference*, 1993.

[HCV52]   David Hilbert and S. Cohn-Vossen. *Geometry and the imagination.* Chelsea Pub. Co., New York, 1952.

[HDD$^+$92]   Hugues Hoppe, Tony DeRose, Tom Duchamp, John McDonald, and Werner Stuetzle. Surface reconstruction from unorganized points. In *Computer Graphics Proceedings (SIGGRAPH 92)*, Annual Conference Series, pages 71–78. ACM SIGGRAPH, 1992.

[HDD$^+$93]   Hugues Hoppe, Tony DeRose, Tom Duchamp, John McDonald, and Werner Stuetzle. Mesh optimization. In *Computer Graphics Proceedings (SIGGRAPH 93)*, Annual Conference Series, pages 19–26. ACM SIGGRAPH, 1993.

[Hem92]   Geoffrey Hemion. *The classification of knots and 3-dimensional spaces.* Oxford University Press, Oxford, 1992.

[HFK94]   John C. Hart, George K. Francis, and Louis H. Kauffman. Visualizing quaternion rotation. *ACM Transactions on Graphics*, 13(3):256–276, July 1994.

[HH91]   Andrew J. Hanson and P. A. Heng. Visualizing the fourth dimension using geometry and light. In *Proceedings of the IEEE Visualization '91 Conference*, San Diego, C.A., 1991.

[HH92]   Andrew J. Hanson and P. A. Heng. Illuminating the fourth dimension. *IEEE Computer Graphics & Applications*, 12(4):54–62, July 1992.

[HK96]   Milana Huang and Louis Kauffman. In *Topology and Geometry in Polymer Science*, University of Minnesota, Minneapolis, June 1996. Institute for Mathematics and its Applications.

[HKG96]   Milana Huang, Louis H. Kauffman, and Robert P. Grzeszczuk. Untangling knots by stochastic energy optimization. In *IEEE Visualization 1996*, pages 279–286, October 1996.

[HLP97]   Joel Haas, Jeffrey C. Lagarias, and Nicholas Pippenger. The computational complexity of knot and link problems. In *Proceedings of the IEEE Symposium of Foundations of Computer Science*, Volume 38, pages 172–181, October 1997.

[Hol88]   Philip Holmes. Knots and orbit genealogies in nonlinear oscillators. In T. Bedford and J. Swift, editors, *New Directions in Dynamical Systems*, Volume 127 of *London Mathematical Society Lecture Note Series*, pages 150–191. Cambridge University Press, 1988.

[Hos97]   Jim Hoste. In Jeff Weeks, editor, *Workshop on Computational Methods in Three Dimensional Topology*, Berkeley, California, March 1997. Mathematical Sciences Research Center.

[HTW98]   Jim Hoste, Morwen B. Thistlethwaite, and Jeff Weeks. The first 1,701,936 knots. *Mathematical Intelligencer*, 1998. (to appear).

[Hug92]   John F. Hughes. Scheduled fourier volume morphing. In *Computer Graphics Proceedings (SIGGRAPH 92)*, Annual Conference Series, pages 43–46. ACM SIG-GRAPH, 1992.

[Hun96]   Kenny Hunt. Ked, 1996. Software for editing knots developed at the University of Iowa.

[HW92]   Shawn R. Henry and Jeffrey R. Weeks. Symmetry groups of hyperbolic knots and links. *Journal of Knot Theory and Its Ramifications*, 1(2):185–201, 1992.

[HZ90]   Christoph M. Hoffmann and Jianhua Zhou. Visualization of surfaces in four-dimensional space. Technical Report CSD-TR-960, Computer Sciences Department, Purdue University, March 1990.

[Jac75]   John David Jackson. *Classical electrodynamics*. Wiley, New York, 2nd edition, 1975.

[Jac90]   E. Atlee Jackson. *Perspectives of nonlinear dynamics*. Cambridge University Press, Cambridge, 1990.

[Joh66]   Norman W. Johnson. Convex polyhedra with regular faces. *Canadian Journal of Mathematics*, 18:169–200, 1966.

[Jon85]   Vaughan F. R. Jones. A polynomial invariant for links via von Neumann algebras. *Bulletin of the American Mathematical Society*, 12:103–112, 1985.

[Jon90]   Vaughan F. R. Jones. Knot theory and statistical mechanics. *Scientific American*, 263(5):98–103, November 1990.

[JS94]   Pawel T. Jochym and Krzysztof Sokalski. Evolution of Hopf configuration of spin field in the Heisenberg model. *J. Phys. A: Math. Gen.*, 27:1353–1361, 1994.

[Kau87]   Louis H. Kauffman. *On Knots*. Princeton University Press, 1987.

[Kau89]   Louis H. Kauffman. Invariants of graphs in three-space. *Transactions of the American Mathematical Society*, 311(2):697–710, February 1989.

[Kau91]   Louis H. Kauffman. *Knots and Physics*, Volume 1 of *Series on Knots and Everything*. World Scientific, Singapore, 1991.

[Kaw96]   Akio Kawauchi. *A Survey of Knot Theory*. Birkhäuser Verlag, Basel, 1996.

[KBBL86]   Hüseyin Koçak, Frederic Bisshopp, Thomas Banchoff, and David Laidlaw. Topology and mechanics with computer graphics. *Advances in Applied Mathematics*, 7:282–308, 1986.

[KBM⁺96]   Vsevolod Katritch, Jan Bednar, Didier Michoud, Robert G. Scharein, Jacques Dubochet, and Andrzej Stasiak. Geometry and physics of knots. *Nature*, 384(6605):142–145, 14 November 1996.

[KG97]   Mario A. Kasapi and John M. Gosline. Design complexity and fracture control in the equine hoof wall. *Journal of Experimental Biology*, 200(11):1639–1659, June 1997. (contains figures generated using KnotPlot).

[KGV84]   S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220:671–680, 1984.

[Kim78]   Scott E. Kim. An impossible four-dimensional illusion. In Brisson [Bri78a], Chapter 11, pages 187–239.

[Kir8̄4]   Thomas P. Kirkman. The enumeration, description, and construction of knots of fewer than ten crossings. *Trans. Royal Soc. Edinburgh*, 32:281–309, 1884.

[Kir8̄5]   Thomas P. Kirkman. The 364 unifilar knots of ten crossings, enumerated and described. *Trans. Royal Soc. Edinburgh*, 32:483–491, 1885.

[Klo86]   Fopke Klok. Two moving coordinate frames for sweeping a 3D trajectory. *Computer Aided Geometric Design*, 3(3):217–229, November 1986.

[Knu86]   Donald E. Knuth. *The* METAFONT*book*. The American Mathematical Society and Addison Wesley, Reading, Massachusetts, 1986.

[KOP⁺97]   Vsevolod Katritch, Wilma K. Olson, Piotr Pieranski, Jacques Dubochet, and Andrzej Stasiak. Properties of ideal composite knots. *Nature*, 388(6638):148, 10 July 1997.

[KP90]   Yan Ke and E. S. Panduranga. A journey into the fourth dimension. In *Proceedings of the IEEE Visualization '90 Conference*, 1990.

[KS94]   Robert B. Kusner and John M. Sullivan. Möbius energies for knots and links, surfaces and submanifolds. Technical Report GCG64, Geometry Center, University of Minnesota, January 1994.

[KW77]   M. Kervaire and C. Weber. A survey of multidimensional knots. In Jean-Claude Hausmann, editor, *Knot Theory*, pages 62–134, Berlin, 1977. Springer-Verlag. Lecture Notes in Mathematics Volume 685.

[Lan85]   John Lankford, circa 1985. Private communication, upon many fascinating walks in the University Endowment Lands.

[Law96]   Ruth J. Lawrence. An introduction to topological field theory. In Louis H. Kauffman, editor, *The Interface of Knots and Physics*, Volume 51 of *Proceedings of Symposia in Applied Mathematics*, pages 89–128, Providence, Rhode Island, 1996. American Mathematical Society.

[Lee74]   Joseph Leeming. *Fun with String*. Dover, New York, 1974. Unabridged republication of the work originally published in 1940 by J. B. Lippincott Company.

[LGB96]   Daniel Lewin, Orli Gan, and Alfred Bruckstein. Trivial or knot: A software tool and algorithms for knot simplification. Preprint, 1996.

[Lit$\overline{85}$]   C. N. Little. On knots, with a census for order ten. *Trans. Conn. Acad. Sci.*, 18:374–378, 1885.

[Lit$\overline{89}$]   C. N. Little. Non-alternate ± knots, of orders eight and nine. *Trans. Royal Soc. Edinburgh*, 35:663–664, 1889.

[Lit$\overline{90}$]   C. N. Little. Alternate ± knots or order 11. *Trans. Royal Soc. Edinburgh*, 36:253–255, 1890.

[Lit00]   C. N. Little. Non-alternate ± knots. *Trans. Royal Soc. Edinburgh*, 39:771–778, 1900.

[Liv93]   Charles Livingston. *Knot Theory*, Volume 24 of *The Carus Mathematical Monographs*. The Mathematical Association of America, Washington, D.C., 1993.

[LM88]   W. B. R. Lickorish and K. C. Millett. The new polynomial invariants of knots and links. *Mathematics Magazine*, 61(1):3–23, February 1988.

[Lom83]   S. J. Lomonaco, Jr. Five dimensional knot theory. *Contemporary Mathematics*, 20:249–270, 1983.

[Lom96]   S. J. Lomonaco, Jr. The modern legacies of Thomson's atomic vortex theory in classical electrodynamics. In Louis H. Kauffman, editor, *The Interface of Knots and Physics*, Volume 51 of *Proceedings of Symposia in Applied Mathematics*, pages 145–166, Providence, Rhode Island, 1996. American Mathematical Society.

[Lor63]   E. N. Lorenz. Deterministic non-periodic flow. *Journal of Atmospheric Science*, 20:130–141, 1963.

[LS94]   Terry J. Ligocki and James A. Sethian. Recognizing knots using simulated annealing. *Journal of Knot Theory and Its Ramifications*, 3(4):477–495, 1994.

[Lyo92]   Kelly A. Lyons. Cluster busting in anchored graph drawing. In John Botsford, Arthur Ryman, Jacob Slonim, and David Taylor, editors, *CASCON '92 — Proceedings of the 1992 CAS Conference, Volume 1*, Toronto, November 1992.

[Max77]   Nelson Max. Turning a sphere inside out. International Film Bureau, Chicago., 1977.

[Mee91]   Aidan Meehan. *Knotwork — The Secret Method of the Scribes*. Thames and Hudson, London, 1991.

[Mei96]   Monica Meissen. Edge numbers of knots, 1996. World Wide Web site http://www.math.uiowa.edu/≈meissen/

[Mil50]   J. Milnor. On the total curvature of knots. *Ann. Math.*, 52:248–257, 1950.

[Mil92]   Kenneth C. Millett. Knot theory, Jones' polynomials, invariants of 3-manifolds, and the topological theory of fluid dynamics. In Moffatt [Mof92b], pages 29–64.

[Mil96]   Kenneth C. Millett. Random walks in polygonal knot space. In *Topology and Geometry in Polymer Science*, University of Minnesota, Minneapolis, June 1996. Institute for Mathematics and its Applications.

[MM$\overline{87}$]   Albert A. Michelson and Edward W. Morley. On the Relative Motion of the Earth and the Luminiferous Ether. *American Journal of Science (Third Series)*, 34(203):333–345, November 1887.

[MMHB93] M. R. Muldoon, R. S. MacKay, J. P. Huke, and D. S. Broomhead. Topology from time series. *Physica D*, 65:1–16, 1993.

[Mof85]   H. K. Moffatt. Magnetostatic equilibria and analogous Euler flows of arbitrarily complex topology. Part 1. Fundamentals. *J. Fluid Mech.*, 159:359–378, 1985.

[Mof90]   H. K. Moffatt. The energy spectrum of knots and links. *Nature*, 347:367–369, September 1990.

[Mof92a]   H. K. Moffatt. Relaxation under topological constraints. In Moffatt [Mof92b], pages 3–28.

[Mof92b]   H. K. Moffatt, editor. *Topological Aspects of the Dynamics of Fluids and Plasmas*. Kluwer Academic, 1992.

[Mon96]   M.B. Monagan. *Maple V programming guide*. Springer, New York, 1996.

[Mor83]   Siegfried Moran. *The Mathematical Theory of Knots and Braids*. Number 82 in North-Holland Mathematics Series. North-Holland, 1983.

[MS92]   Henry P. Moreton and Carlo H. Sequin. Functional optimization for fair surface design. In *Computer Graphics Proceedings (SIGGRAPH 92)*, Annual Conference Series, pages 167–176. ACM SIGGRAPH, 1992.

[MSS97]   Chengde Mao, Weiqiong Sun, and Nadrian C. Seeman. Assembly of Borromean rings from DNA. *Nature*, 386(6621):137–138, 13 March 1997.

[MTW73]   Charles W. Misner, Kip S. Thorne, and John Archibald Wheeler. *Gravitation*. W. H. Freeman, San Francisco, 1973.

[Mur96]   Kunio Murasugi. *Knot Theory and Its Applications*. Birkhäuser, Boston, 1996. Published originally in 1993 in Japanese as *Musubime riron to sono ōyō*.

[MW88]   Matthew Moore and Jane Wilhelms. Collision detection and response for computer animation. In *Computer Graphics Proceedings (SIGGRAPH 88)*, Annual Conference Series, pages 289–298. ACM SIGGRAPH, 1988.

[Nak93]   Nobumitsu Nakauchi. A remark on O'Hara's energy of knots. *Proceedings of the American Mathematical Society*, 118(1), May 1993.

[NDW93]   Jackie Neider, Tom Davis, and Mason Woo. *OpenGL Programming Guide*. Addison-Wesley, Reading, Massachusetts, 1993.

[Neg91]   S. Negami. Ramsey theorems for knots, links, and spatial graphs. *Transactions of the American Mathematical Society*, 324:527–541, 1991.

[Neu79]   Lee Neuwirth. The theory of knots. *Scientific American*, 240(6):110–124, June 1979.

[New96]   William H. New. *Science Lessons*. Oolichan Books, Lantzville, British Columbia, 1996.

[Nol78]   A. Michael Noll. Displaying n-dimensional hyperobjects by computer. In Brisson [Bri78a], Chapter 7, pages 147–158.

[Nol94]   A. Michael Noll. The beginnings of computer art in the United States: A memoir. *Leonardo*, 27(1):39–44, 1994.

[Och90]   Mitsuyuki Ochiai. Non-trivial projections of the trivial knot. In *S.M.F. Astérisque*, Volume 192, pages 7–10. Société Mathématique de France, 1990.

[O'H91]   Jun O'Hara. Energy of a knot. *Topology*, 30(2):241–247, 1991.

[O'H92]   Jun O'Hara. Family of energy functionals of knots. *Topology and its Applications*, 48:147–161, 1992.

[Per74]   Kenneth A. Perko. On the classifications of knots. *Proceedings of the American Mathematical Society*, 45:262–266, 1974.

[Pet88]   Ivars Peterson. Unknotting a tangled tale. *Science News*, 133:328–330, 21 May 1988.

[PFTV88]   William H. Press, Brian P. Flannery, Saul A. Teukolsky, and William T. Vetterling. *Numerical Recipes in C*. Cambridge University Press, Cambridge, 1988.

[PG92]   Mark Phillips and Charlie Gunn. Visualizing hyperbolic space: Unusual uses of 4×4 matrices. In David Zeltzer, editor, *Computer Graphics Special Issue (1992 Symposium on Interactive 3D Graphics)*, Volume 26, pages 209–214, March 1992.

[Phi66]   Anthony Phillips. Turning a sphere inside out. *Scientific American*, 214(5):112–120, May 1966.

[Pip89]   Nicholas Pippenger. Knots in random walks. *Discrete Applied Mathematics*, 25:273–278, 1989.

[PLM93]   Mark Phillips, Silvio Levy, and Tamara Munzner. GeomView — an interactive geometry viewer. *Notices of the American Mathematical Society*, October 1993.

[PT87]   Jósef H. Przytycki and Paweł Traczyk. Invariants of links of the Conway type. *Kobe Journal of Mathematics*, 4(2):115–139, 1987.

[Ran94]   Richard Randell. An elementary invariant of knots. *Journal of Knot Theory and Its Ramifications*, 3(3):279–286, 1994.

[Ran96]   Richard Randell, 1996. Private communication.

[Rav89]   Douglas C. Ravenel. Homotopy groups of spheres on a small computer. In Martin C. Tangora, editor, *Computers in Geometry and Topology*, Volume 114 of *Lecture Notes in Pure and Applied Mathematics*, Chapter 12, pages 259–284. Marcel Dekker, Inc., New York, 1989.

[Rea79]   Ronald C. Read. Algorithms in graph theory. In Robin J. Wilson and Lowell W. Beineke, editors, *Applications of Graph Theory*, Chapter 13, pages 381–417. Academic Press, London, 1979.

[Rei35]   K. Reidemeister. *Knot Theory*. BCS Associates, Moscow, Idaho, 1935.

[Rei57]   Hans Reichenbach. *The Philosophy of Space and Time*. Dover, New York, 1957. English translation of the German original *Philosophie der Raum-Zeit-Lehre* (1927).

[Rol76]   Dale Rolfsen. *Knots and Links*. Publish or Perish, Inc., 1976.

[Rol94]   Dale Rolfsen. Global mutation of knots. *Journal of Knot Theory and Its Ramifications*, 3(3):407–417, 1994.

[Ros89]   Dennis Roseman. Spinning knots about submanifolds: Spinning knots about projections of knots. *Topology and its Applications*, 31:225–241, 1989.

[Ros92a]   Dennis Roseman. Design of a mathematicians' drawing program. In Steve Cunningham, Nancy Knolle Craighill, Martin W. Fong, and Judith R. Brown, editors, *Computer Graphics Using Object-Oriented Programming*, Chapter 11, pages 279–295. John Wiley & Sons, New York, 1992.

[Ros92b]   Dennis Roseman. Motions of flexible objects. In Tosiyasu L. Kunii and Yoshihisa Shinagawa, editors, *Modern Geometric Computing for Visualization*, pages 91–120. Springer-Verlag, Tokyo, New York, 1992.

[Ros95]   Dennis Roseman. Wiener's thought on the computer as an aid in visualizing higher-dimensional forms. Preprint, February 1995.

[Ros96]   Dennis Roseman. What should a surface in 4-space look like? Preprint, January 1996.

[SB84]   Uri Shani and Dana H. Ballard. Splines as embeddings for generalized cylinders. *Computer Vision, Graphics, and Image Processing*, 27(2):129–156, August 1984.

[Sch53]   H. Schubert. Construction of a satellite. *Acta Math.*, 90:131, 1953.

[Sch90]   Bernard F. Schutz. *Geometrical methods of mathematical physics*. Cambridge University Press, Cambridge, U.K., 1990.

[Sch91]   Robert G. Scharein. Computer program visualizes knots. *Campus Computing*, pages 5–6, October 1991. University of British Columbia Computing Services' monthly newsletter.

[Sch92]   Martin Scharlemann. Topology of knots. In Moffatt [Mof92b], pages 65–82.

[Sch95]   Robert G. Scharein, 1995. Computer graphics for the University of British Columbia Continuing Studies Calender.

[Sch96]   Robert G. Scharein. Visualizing knots with KnotPlot. In *Topology and Geometry in Polymer Science*, University of Minnesota, Minneapolis, June 1996. Institute for Mathematics and its Applications.

[Sch97a]   Robert G. Scharein, April 1997. Computer graphics for special issue on knots entitled *La Science des Nœuds* appearing in *Dossier Pour la Science* (Édition Française de Scientific American), pages 5, 41, 47, and 119.

[Sch97b]   Robert G. Scharein. *KnotPlot — A Program for Viewing Mathematical Knots (User Manual)*, 1997. Also available from the KnotPlot Site at the World Wide Web site http://www.cs.ubc.ca/nest/imager/contributions/scharein/KnotPlot.html

[Sed96]   Eric Sedgwick. *pretzel*. Video tape illustration of the proof of a theorem in [Sed97], September 1996. Information available at the World Wide Web site http://www.ma.utexas.edu/users/sedgwick/pretzel/pretzel.html

[Sed97] Eric Sedgwick. An infinite collection of Heegaard splittings that are equivalent after one stabilization. *Mathematische Annalen*, 308(1):65–72, 1997.

[Sho92] Ken Shoemake. ARCBALL: A user interface for specifying three-dimensional orientation using a mouse. In *Proceedings of Graphics Interface '92*, pages 151–156, May 1992. held in Vancouver, B.C.; 11-15 May 1992.

[Sim87] Jonathan K. Simon. A topological approach to the stereochemistry of nonrigid molecules. 1987.

[Sim94] Jonathan K. Simon. Energy functions for polygonal knots. *Journal of Knot Theory and Its Ramifications*, 3(3):299–320, 1994.

[Sim96] Jonathan Simon. Energy and thickness of knots. In *Topology and Geometry in Polymer Science*, University of Minnesota, Minneapolis, June 1996. Institute for Mathematics and its Applications.

[Sno90] Jack Snoeyink. A trivial knot whose spanning disks have exponential size. In *Proceedings of the Sixth Annual Symposium on Computational Geometry*, pages 139–147, Berkeley, California, 6–8 June 1990. Association for Computing, ACM Press.

[Sul93] Mike Sullivan. Prime decomposition of knots in Lorenz-like templates. *Journal of Knot Theory and Its Ramifications*, 2(4):453–462, 1993.

[Sum85] D. W. Sumners. The role of knot theory in DNA research. In Clint McCrory and Theodore Shifrin, editors, *Geometry and Topology — Manifolds, Varieties, and Knots*, Volume 105 of *Lecture Notes in Pure and Applied Mathematics*, pages 297–318. Marcel Dekker, Inc., 1985.

[Sur92a] Mark C. Surles. An algorithm with linear complexity for interactive, physically-based modeling of large proteins. In *Computer Graphics Proceedings (SIGGRAPH 92)*, Annual Conference Series, pages 221–230. ACM SIGGRAPH, 1992.

[Sur92b] Mark C. Surles. Interactive modeling enhanced with constraints and physics with applications in molecular modeling. In David Zeltzer, editor, *Computer Graphics Special Issue (1992 Symposium on Interactive 3D Graphics)*, Volume 26, pages 175–182, March 1992.

[Tai76] Peter Guthrie Tait. On knots. *Trans. Royal Soc. Edinburgh*, 1876. Reprinted as [Tai98].

[Tai84] Peter Guthrie Tait. On knots. Part II. *Trans. Royal Soc. Edinburgh*, 1884. Reprinted as [Tai98].

[Tai85] Peter Guthrie Tait. On knots. Part III. *Trans. Royal Soc. Edinburgh*, 1885. Reprinted as [Tai98].

[Tai98]   Peter Guthrie Tait. On knots I. II. III. In *Scientific Papers, I. 1877–1885*, pages 273–437. Cambridge University Press, Cambridge, 1898.

[Thi85]   Morwen B. Thistlethwaite. Knot tabulations and related topics. In I. M. James and E. H. Kronheimer, editors, *Aspects of Topology in Memory of Hugh Dowker*, Volume 93 of *London Mathematical Society Lecture Note Series*, pages 1–76. Cambridge University Press, 1985.

[Thi97]   Morwen B. Thistlethwaite. In Jeff Weeks, editor, *Workshop on Computational Methods in Three Dimensional Topology*, Berkeley, California, March 1997. Mathematical Sciences Research Center.

[Tho67]   Sir William Thomson. On vortex atoms. *Proc. Royal Soc. Edinburgh*, 6:94–105, 1867. Also appears on pages 1–12 of [Tho10].

[Tho69]   Sir William Thomson. On vortex motion. *Trans. Royal Soc. Edinburgh*, 25:217–260, 1869. Also appears on pages 13–66 of [Tho10].

[Tho75]   Sir William Thomson. Vortex statics. *Proc. Royal Soc. Edinburgh*, 1875. Also appears on pages 115–128 of [Tho10].

[Tho10]   Sir William Thomson. *Mathematical and Physical Papers*, Volume 4. Cambridge University Press, Cambridge, 1910.

[Tut63]   W. T. Tutte. How to draw a graph. *Proc. London Math. Soc.*, 13:743–768, 1963.

[TvdG96]   John C. Turner and Pieter van de Griend, editors. *History and Science of Knots*, Volume 11 of *Series on Knots and Everything*. World Scientific, Singapore, 1996.

[vdG92]   Pieter van de Griend. *Knots and Rope Problems*. Privately published in a limited edition, Århus, September 1992.

[vdG93]   Pieter van de Griend. *Notes on Knots*. Privately published in one limited edition of 50 copies, Århus, March 1993.

[Wee90]   Jeff Weeks. *Experimental Mathematics*, 1990.

[Wel94]   William Welch. *Serious Putty: Topological Design for Variational Curves and Surfaces*. PhD thesis, School of Computer Science, Carnegie Mellon University, 1994. Published as technical report CMU-CS-95-217.

[Whi33]   H. Whitney. A set of topological invariants for graphs. *American Journal of Mathematics*, 55:321–335, 1933.

[Whi37]   J. H. C. Whitehead. On doubled knots. *J. London Math. Soc.*, 12:63–71, 1937.

[Wil83]   R. F. Williams. Lorenz knots are prime. *Ergodic Theory and Dynamical Systems*, 4:147–163, 1983.

[Wol91]   Stephen Wolfram. *Mathematica — A System for Doing Mathematics by Computer*. Addison-Wesley, Redwood City, California, 2nd edition, 1991.

[Wu92]   F. Y. Wu. Knot theory and statistical mechanics. *Reviews of Modern Physics*, 64(4):1099–1131, October 1992.

[Wu96]   Ying-Qing Wu. MING – an MD knot energy minimizing program. In *Topology and Geometry in Polymer Science*, University of Minnesota, Minneapolis, June 1996. Institute for Mathematics and its Applications.

[WW92]   William Welch and Andrew Witkin. Variational surface modeling. In *Computer Graphics Proceedings (SIGGRAPH 92)*, Annual Conference Series, pages 157–166. ACM SIGGRAPH, 1992.

[Zal69]   Viktor A. Zalgaller. *Convex Polyhedra with Regular Faces*. 1969.

[Zee62]   E. C. Zeeman. Isotopies and knots in manifolds. In Fort Jr. [For62], Chapter 3, pages 187–193.

Appendix A

Knot catalogues

A.1 Minimal stick candidates

A.1.1 Stereoscopic pairs

This section contains stereoscopic images of minimal stick representatives and candidates for all prime knots up to eight crossings. Several other knots of interest and a few links are also included. All knots are in minimal energy positions for the MD-energy, E_{MD} and are aligned so that the line of sight is along the principal moment of inertia axis with the largest eigenvalue. See Section 3.5.1 for a definition of E_{MD}. For each knot, the number of sticks is given below the knot name. A star is used to indicate that the value is the stick-number of the knot (so the knot shown is a representative).

To accommodate both "wide-eyed" and "cross-eyed" viewers, no occlusions are drawn in the figures. To see the stereoscopic effect in "wide-eyed" mode, relax your eyes and fix your gaze as if viewing a distant object. Shifting your attention to the paper, you should notice multiple, out of focus images. Eventually, you should be able to merge the centre two images into one. After a bit of concentration, your eyes will shift into focus on the paper, and the image will appear to pop out. "Cross-eyed" viewing is similar, except that the right eye focuses on the left-hand image, and the left eye on the right-hand. You may find your finger useful in helping to initially cross your eyes. For best results in both cases, hold the paper about 30cm from your eyes in good lighting conditions.

The minimal E_{MD} configurations reveal interesting symmetries of several knots, in particular 7_1 (the torus knot $K_{2,7}$) and 8_5, and some links such as 6_1^2, 6_2^3 (the Borromean rings), and 8_7^2. Many links (7_7^2 is an example) have a stick number that is trivially the sum of the stick numbers of its components. These have been avoided as far as is obvious, however, it is possible that by rearrangement some of the links shown may be brought into a configuration where this would be true.

Appendix B gives a location on the World Wide Web where all the coordinate data for these knots may be found.
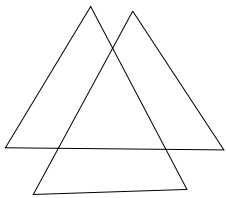
3_1
$s = 6^\star$
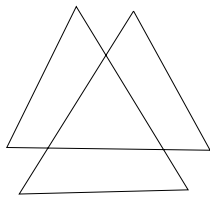
$E_{MD} = 189.9$
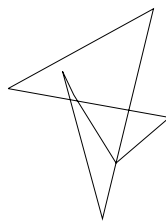
4_1
$s = 7^\star$
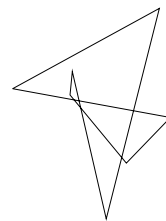
$E_{MD} = 459.6$
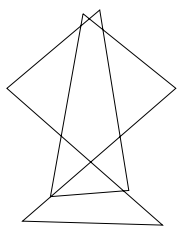
5_1
$s = 8^\star$
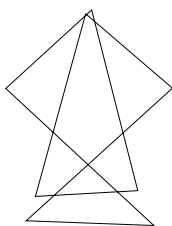
$E_{MD} = 401.4$
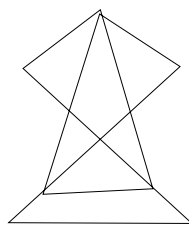
5_2
$s = 8^\star$
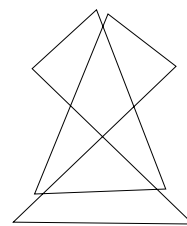
$E_{MD} = 415.6$
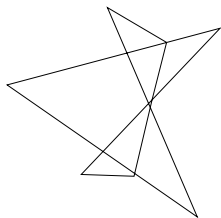
6_1
$s = 8^\star$
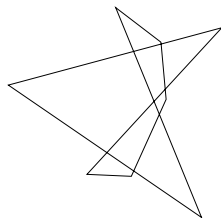
$E_{MD} = 1047.4$
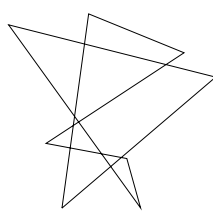
6_2
$s = 8^\star$
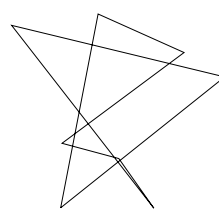
$E_{MD} = 879.2$
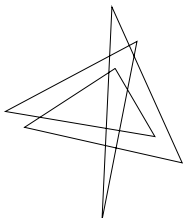
6_3
$s = 8^\star$
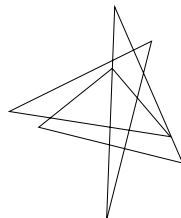
$E_{MD} = 1133.8$
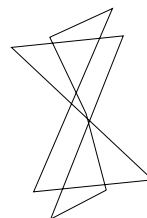
7_1
$s = 9$
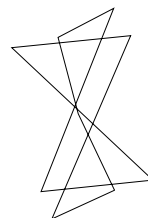
$E_{MD} = 836.7$
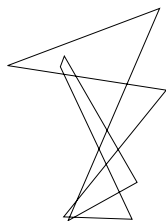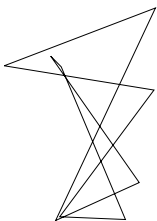
7_2
$s = 9$
$E_{MD} = 1116.0$
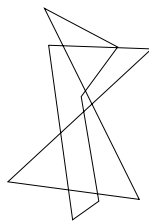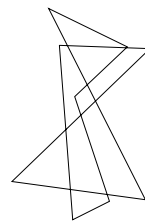
7_3
$s = 9$
$E_{MD} = 808.7$
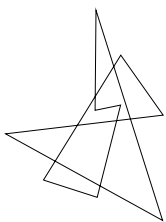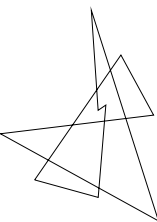
7_4
$s = 9$
$E_{MD} = 1074.6$
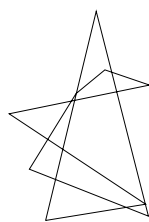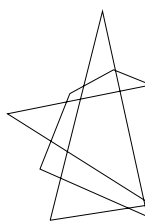
7_5
$s = 9$
$E_{MD} = 1105.3$
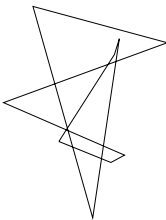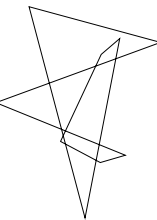
7_6
$s = 9$
$E_{MD} = 1218.4$
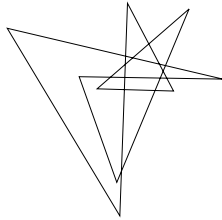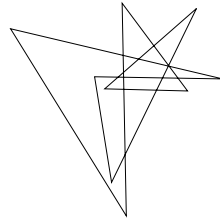
7_7
$s = 9$
$E_{MD} = 1288.9$
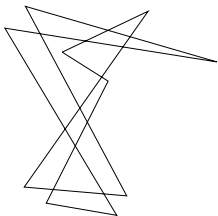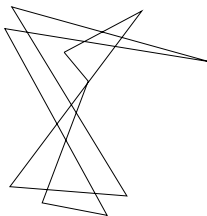
8_1
$s = 10$
$E_{MD} = 1316.5$
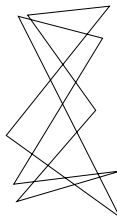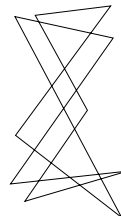
8_2
$s = 10$
$E_{MD} = 992.7$
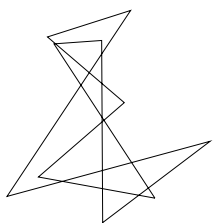
8_3
$s = 10$
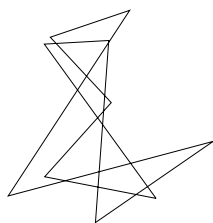
$E_{MD} = 1203.3$
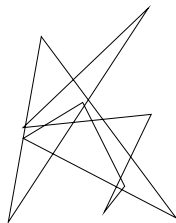
8_4
$s = 10$
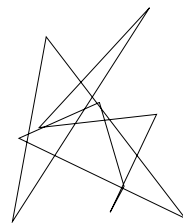
$E_{MD} = 1731.6$
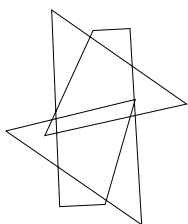
8_5
$s = 10$
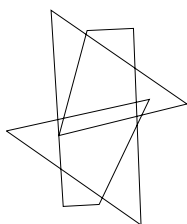
$E_{MD} = 1135.7$
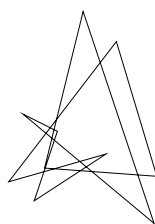
8_6
$s = 10$
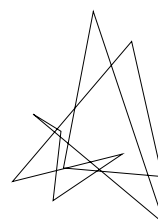
$E_{MD} = 1732.1$
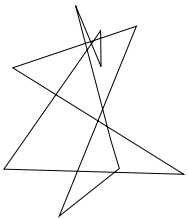
8_7
$s = 10$
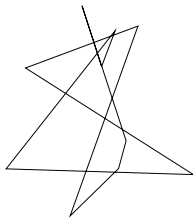
$E_{MD} = 1159.3$
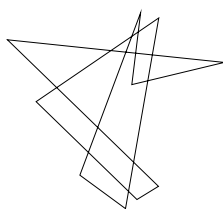
8_8
$s = 10$
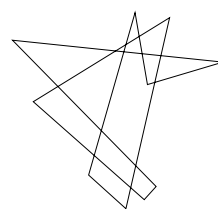
$E_{MD} = 1323.4$
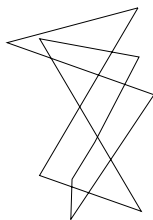
8_9
$s = 10$
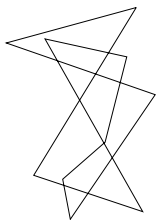
$E_{MD} = 1062.8$

8_{10}
$s = 10$

$E_{MD} = 1642.1$

186
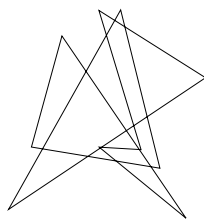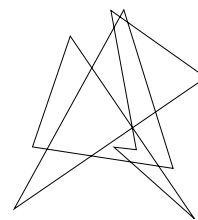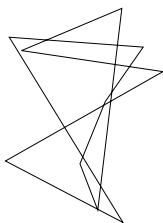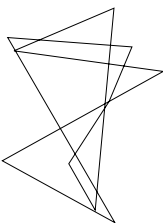
8_{11}
$s = 10$
$E_{MD} = 1811.8$
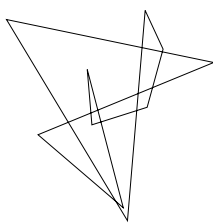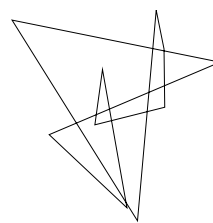
8_{12}
$s = 10$
$E_{MD} = 1785.7$
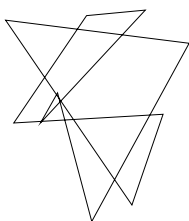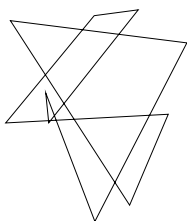
8_{13}
$s = 10$
$E_{MD} = 1224.2$
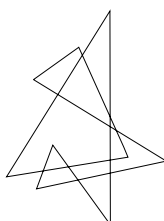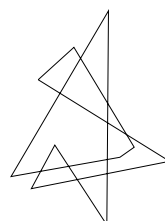
8_{14}
$s = 10$
$E_{MD} = 1453.3$
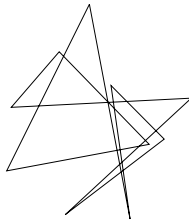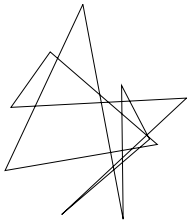
8_{15}
$s = 10$
$E_{MD} = 1766.1$
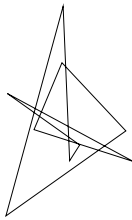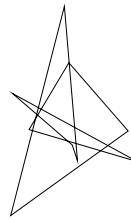
8_{16}
$s = 9$
$E_{MD} = 2265.8$
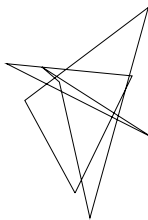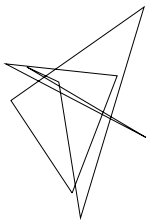
8_{17}
$s = 9$
$E_{MD} = 2539.8$
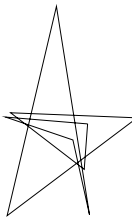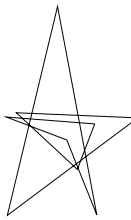
8_{18}
$s = 9$
$E_{MD} = 2923.5$
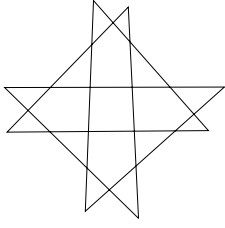
8_{19}
$s = 8^\star$
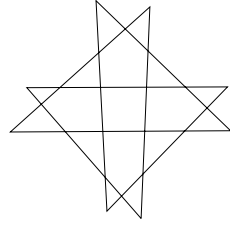
$E_{MD} = 1230.5$
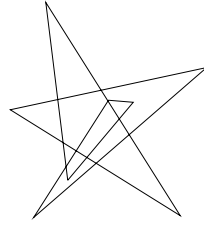
8_{20}
$s = 8^\star$
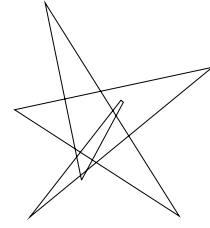
$E_{MD} = 2200.2$
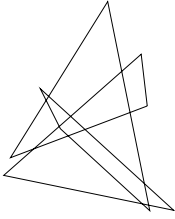
8_{21}
$s = 9$
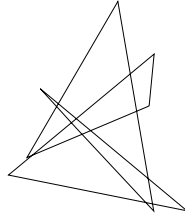
$E_{MD} = 1062.2$
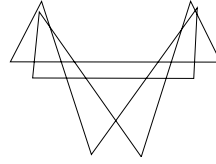
9_1
$s = 10$
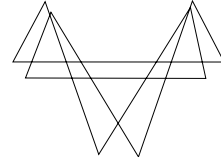
$E_{MD} = 1711.7$
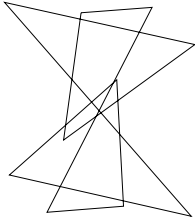
9_{16}
$s = 10$
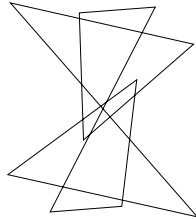
$E_{MD} = 2193.6$
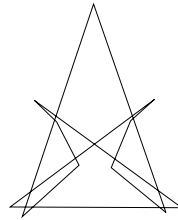
9_{23}
$s = 11$
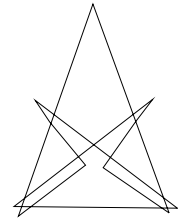
$E_{MD} = 2213.1$
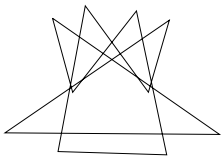
9_{35}
$s = 10$
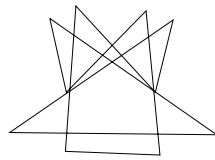
$E_{MD} = 1940.1$
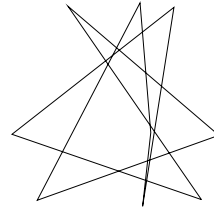
9_{41}
$s = 9$
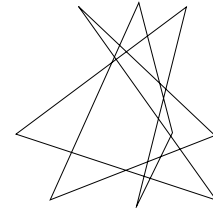
$E_{MD} = 1990.5$
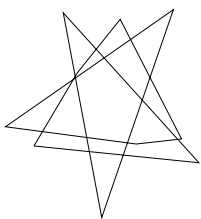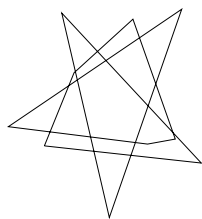
10_{109}
$s = 10$
$E_{MD} = 2477.4$
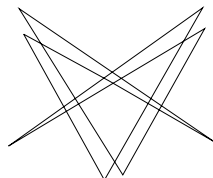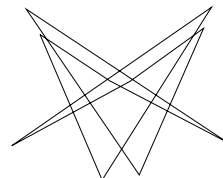
10_{120}
$s = 10$
$E_{MD} = 3489.3$
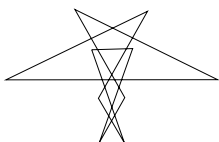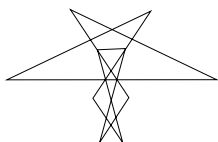
10_{124}
$s = 10$
$E_{MD} = 1157.0$
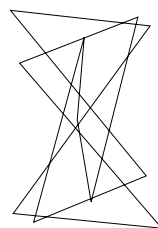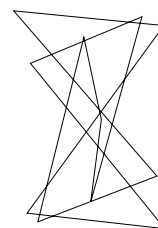
10_{142}
$s = 11$
$E_{MD} = 1133.1$
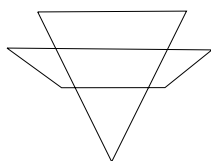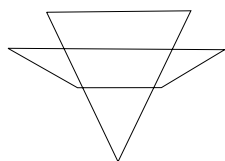
4_1^2
$s = 7$
$E_{MD} = 310.6$
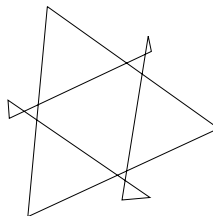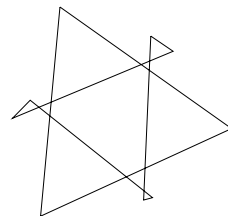
6_1^2
$s = 9$
$E_{MD} = 501.0$
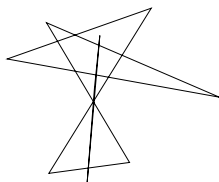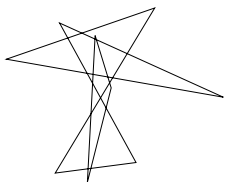
6_2^2
$s = 9$
$E_{MD} = 811.1$
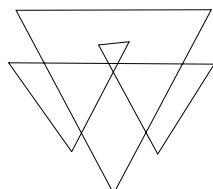
6_3^2
$s = 9$
$E_{MD} = 714.1$
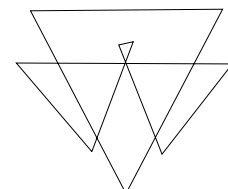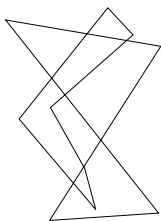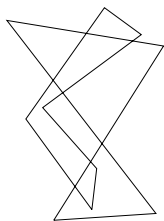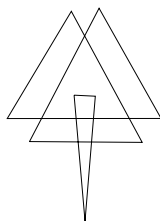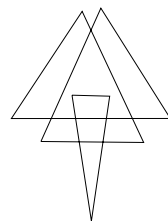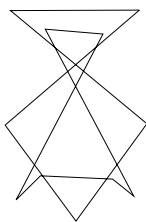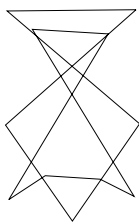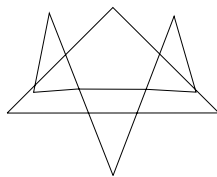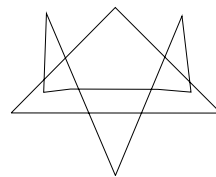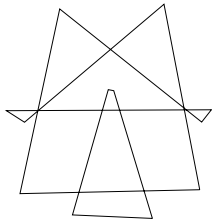
189

7^2_2
$s = 10$
$E_{MD} = 894.1$

7^2_7
$s = 9^\star$
$E_{MD} = 418.0$

8^2_1
$s = 11$
$E_{MD} = 737.9$

8^2_7
$s = 10$
$E_{MD} = 1184.1$

9^2_{13}
$s = 12$
$E_{MD} = 1051.3$

9^2_{33}
$s = 12$
$E_{MD} = 1574.0$

9^2_{46}
$s = 12$
$E_{MD} = 771.5$

9^2_{61}
$s = 10$
$E_{MD} = 893.9$
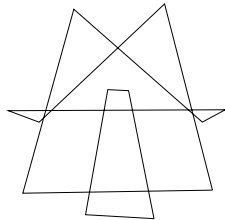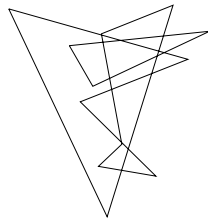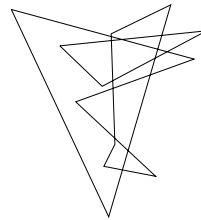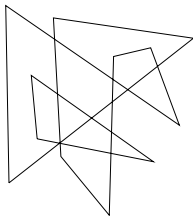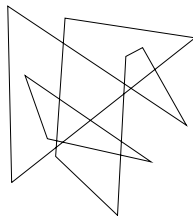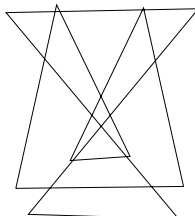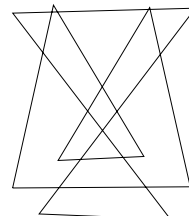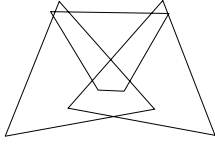
6^3_2
$s = 10^\star$
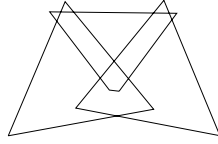
$E_{MD} = 732.1$

6^3_3
$s = 9$

$E_{MD} = 311.1$

9^3_{12}
$s = 11$

$E_{MD} = 2097.8$

9^3_{20}
$s = 11$
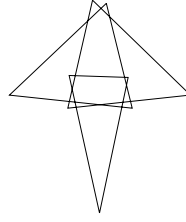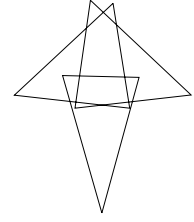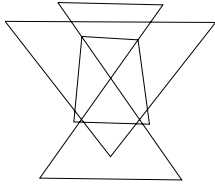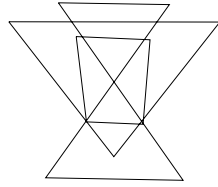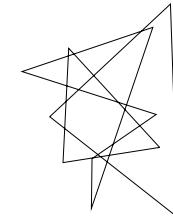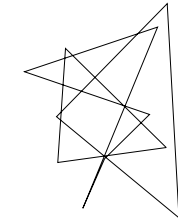
$E_{MD} = 1110.7$
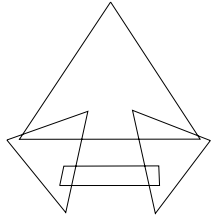
8^4_1
$s = 13^\star$
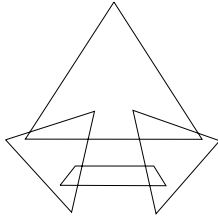
$E_{MD} = 977.9$
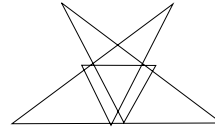
Square
$s = 8^\star$
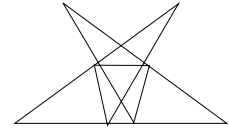
$E_{MD} = 1972.5$
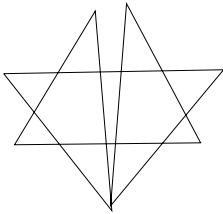
Granny
$s = 8^\star$
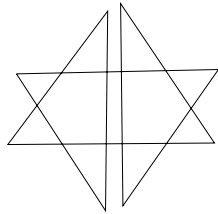
$E_{MD} = 2357.1$
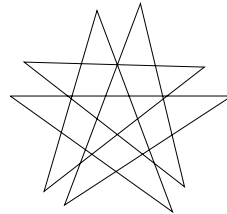
$K_{4,5}$
$s = 10^\star$
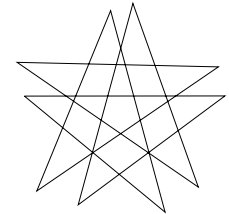
$E_{MD} = 4616.4$

A.1.2 Orthographic projections along principal axes

As in the previous section with the stereoscopic pairs, this section shows minimal stick candidates and representatives in a minimal energy conformation for E_{MD}, except that here are shown orthographic projections along each of the principal axes of inertia. This catalogue includes all knots that exhibit interesting symmetries (either bilateral or two-fold rotational) when projected in this manner.

The knots have been aligned so that the y and z-axes correspond to the principal axes with the smallest and largest eigenvalues, respectively. Numbers in parentheses indicate the ratio of the eigenvalue associated with that axis to the smallest eigenvalue (associated with the y-axis). The error on these ratios is estimated to be ± 0.001, so within experimental error the knots 3_1, 8_{19}, and $K_{4,5}$ have two eigenvalues of equal value, as might be expected. Despite the mystical and historical significance of the Borromean rings (6_2^3), it is probably a coincidence that the ratio of the maximum to minimum eigenvalues of this special conformation is equal to the Golden Ratio (to within experimental error). All the pictures are at the same scale in that the minimum distance between any two non-adjacent edges is the same in each case.

A small figure included by each picture indicates the directions of the remaining axes, as in ⌐, where the filled circle indicates that the x-axis points up out of the page. Similarly, in ⌐ the unfilled circle indicates that the y-axis points down into the page. This is obvious since the x, y, and z-axes form a right-handed triad, however, filling the circles in this way makes it easier to visualize the direction of the remaining axis.

x (2.209)　　　　　y　　　　　z (2.334)　　　7_1

x (1.503)　　　　　y　　　　　z (1.519)　　　8_5

x (1.000)　　　　　y　　　　　z (1.046)　　　8_{19}

193

x (1.533) y z (1.661) 9_1

x (1.368) y z (1.412) 9_{16}

x (1.155) y z (1.578) 9_{23}

194

x (1.096) y z (1.267) 9_{35}

x (1.195) y z (1.415) 10_{120}

x (1.227) y z (1.578) 10_{124}

x (1.692)

y

z (1.712) 10_{142}

x (1.645)

y

z (1.672) 4_1^2

x (1.145)

y

z (1.478) 6_3^2

x (1.490)

y

z (1.640) 7_7^2

196

x (1.763)   y   z (1.916) 8^2_1

x (1.454)   y   z (1.593) 8^2_7

x (1.059)   y   z (1.386) 9^2_{13}

x (1.352)   y   z (1.482) 9^2_{61}

197

x (1.239)　　　　　y　　　　　z (1.618)　　　6_2^3

x (1.326)　　　　　y　　　　　z (1.494)　　　6_3^3

x (1.300)　　　　　y　　　　　z (1.581)　　　8_1^3

x (1.138)　　　　　y　　　　　z (1.467)　　　9_{12}^3

x (1.304)
y
z (1.731)
8_1^4

x (1.312)
y
z (1.405)
Square

x (1.043)
y
z (1.046)
Granny

199

x (1.067)   y   z (1.068) $K_{4,5}$

A.2   Minimal symmetric energy knots and links

Colour Plate 3 shows images of all knots up to eight crossings, relaxed under the symmetric energy, E_S. The knots are visualized as radiating tubes as explained in Section 8.2.3, and normalized such that each knot has the same length. Included with each image is the value of the E_S for that conformation. This data is plotted in Figure 8.7. Images of much higher quality than these here are available at the *Symmetric Energy* World Wide Web site (see Appendix B for the address).

Appendix B

Supplementary material

A World Wide Web site exists that contains pointers to various data and supplementary material of interest to readers of this thesis.

<div align="center">

`http://www.cs.ubc.ca/spider/scharein/thesis.html`

</div>

This site contains vertex data for all the minimal stick candidates, MPEG movies of the knot relaxations shown in the thesis, and a hypertext version of this thesis, as well as a downloadable version in PostScript similar to this version.

High quality images and MPEG movies of knots as radiating tubes are available at the *Symmetric Energy Site*:

<div align="center">

`http://www.cs.ubc.ca/spider/scharein/SE.html`

</div>

All the knot vertex data for the symmetric energy catalogue in Appendix A is available at this site.

Finally, a wide variety of images created with KnotPlot, some mathematical, some "artistic", and some not easily categorized are at the *KnotPlot Site*:

<div align="center">

`http://www.cs.ubc.ca/nest/imager/contributions/scharein/KnotPlot.html`

</div>

Index

$K_{2,2}$ $K_{2,3}$ $K_{2,4}$ $K_{2,5}$

$K_{3,3}$ $K_{2,6}$ $K_{2,7}$ $K_{3,4}$

$K_{2,8}$ $K_{2,9}$ $K_{2,10}$ $K_{3,5}$

$K_{2,11}$ $K_{2,12}$ $K_{4,4}$ $K_{3,6}$

$K_{2,13}$ $K_{2,14}$ $K_{3,7}$ $K_{2,15}$

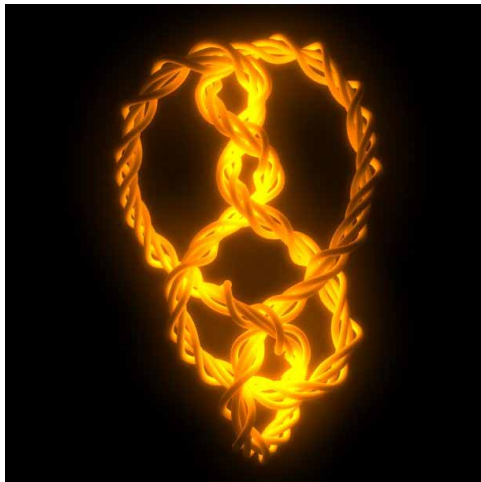$K_{4,5}$ $K_{2,16}$ $K_{3,8}$ $K_{2,17}$

Colour Plate 1: All torus knots and links with crossing number less than 18.
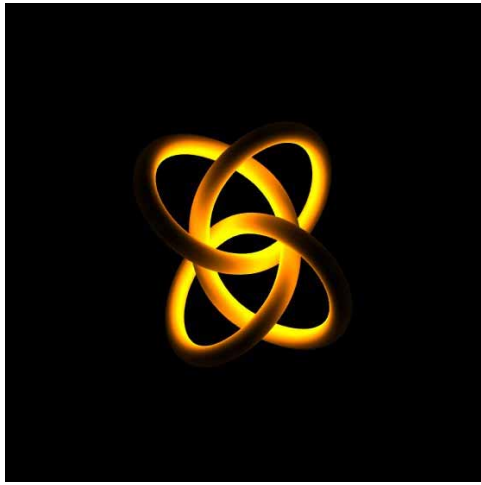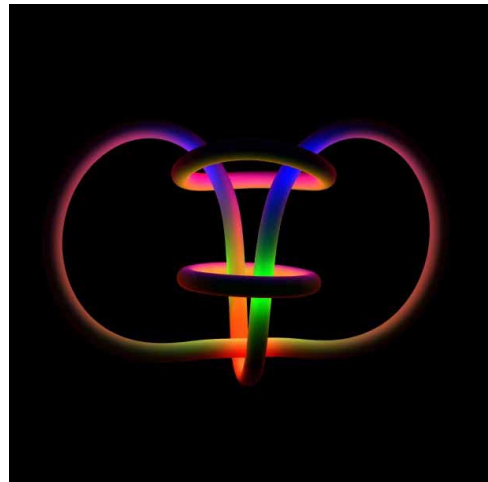
(a)
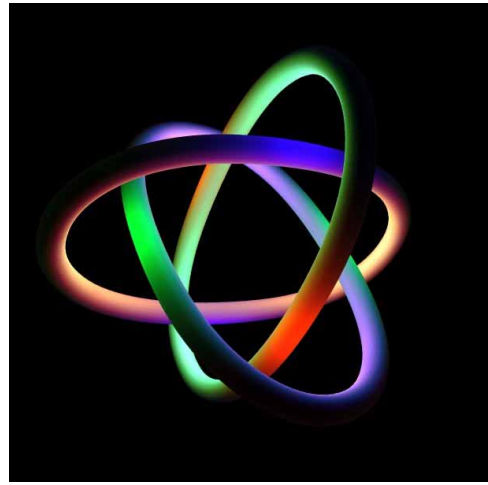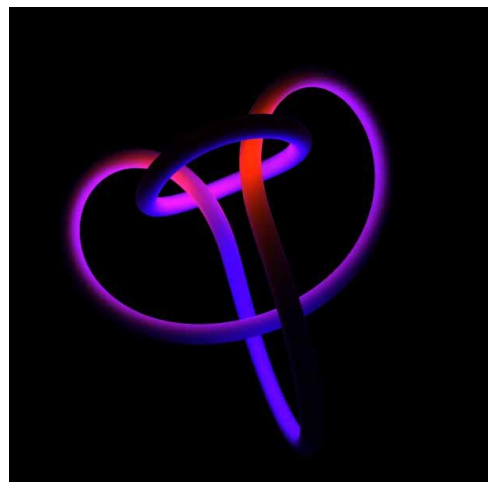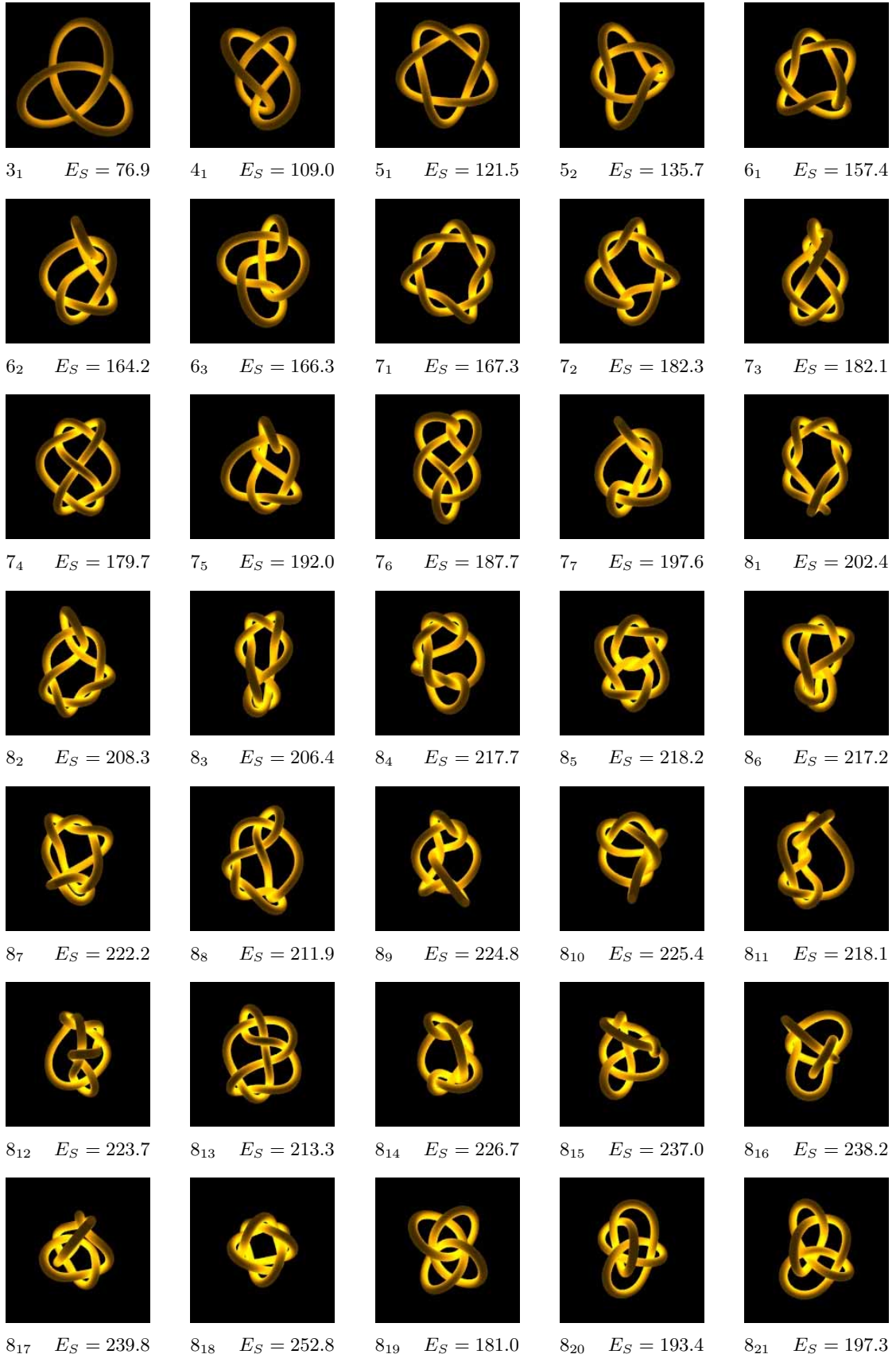
(b)

(c)

(d)
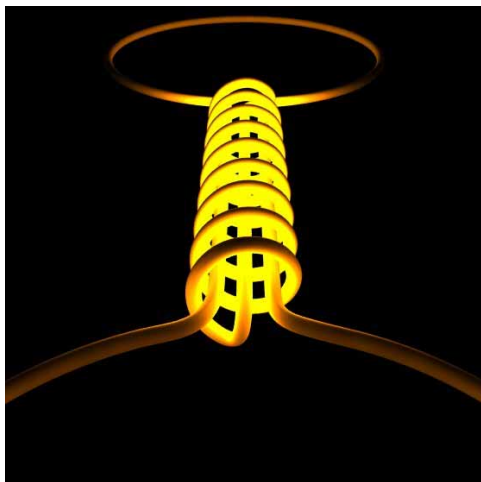
(e)

(f)

Colour Plate 2: Knots as radiating tubes.

3_1 $E_S = 76.9$ 4_1 $E_S = 109.0$ 5_1 $E_S = 121.5$ 5_2 $E_S = 135.7$ 6_1 $E_S = 157.4$

6_2 $E_S = 164.2$ 6_3 $E_S = 166.3$ 7_1 $E_S = 167.3$ 7_2 $E_S = 182.3$ 7_3 $E_S = 182.1$

7_4 $E_S = 179.7$ 7_5 $E_S = 192.0$ 7_6 $E_S = 187.7$ 7_7 $E_S = 197.6$ 8_1 $E_S = 202.4$

8_2 $E_S = 208.3$ 8_3 $E_S = 206.4$ 8_4 $E_S = 217.7$ 8_5 $E_S = 218.2$ 8_6 $E_S = 217.2$

8_7 $E_S = 222.2$ 8_8 $E_S = 211.9$ 8_9 $E_S = 224.8$ 8_{10} $E_S = 225.4$ 8_{11} $E_S = 218.1$

8_{12} $E_S = 223.7$ 8_{13} $E_S = 213.3$ 8_{14} $E_S = 226.7$ 8_{15} $E_S = 237.0$ 8_{16} $E_S = 238.2$

8_{17} $E_S = 239.8$ 8_{18} $E_S = 252.8$ 8_{19} $E_S = 181.0$ 8_{20} $E_S = 193.4$ 8_{21} $E_S = 197.3$

Colour Plate 3: Normalized symmetric energy zoo.