Untangling Knots by Stochastic Energy Optimization

Milana Huang, Robert P. Grzeszczuk†, Louis H. Kauffman*

Electronic Visualization Laboratory
Departments of EECS and *Mathematics
University of Illinois at Chicago

†Department of Radiology
University of Chicago

Abstract

A method for visualizing unknottedness of mathematical knots via energy optimization with simulated annealing is presented. In this method a potential field is formed around a tangled rope that causes it to self-repel. By allowing the rope to evolve in this field in search of an energy minimizing configuration we can determine the knot type of the initial configuration. In particular, it is natural to conjecture that if such a "charged rope" was not initially knotted, it will reach its minimal potential in a circular configuration, given a suitable energy functional. Because situations potentially arise in which the functional may not be strictly unimodal, we suggest it to be advantageous to use a robust stochastic optimization technique (simulated annealing), rather than a deterministic hill climber common in physically-based approaches, to make sure that the evolving rope does not settle in a suboptimal configuration. The same method is applicable to simplifying arbitrary knots and links and for establishing knot equivalence. Aside from its theoretical appeal, the method promises to solve practical problems common in genetic research and polymer design.

1 Introduction

This paper is devoted to visualizing the unknottedness of a complex space curve. In the course of the work, features of space curves (such as their energy distribution and their local stability) are also visualized. The purpose of the method can be simply stated as follows: given a piece of string that was tied in an arbitrary manner and whose ends were then pasted together, to simplify it as much as possible. In particular, we are interested in determining if a particular tangle can be simplified to a circle in which case it can be shown not to have been knotted in the first place. However, the method that we propose is applicable to classifying arbitrary knots and links as well as for establishing knot equivalence in a general case.

Let us define a knot as a closed curve in 3-dimensional space that does not self-intersect and has no thickness. The formal definition of a knot K is $K : S^1 \rightarrow R^3$ where S^1 is the 1-dimensional sphere embedded in R^3, 3-dimensional Euclidean space. We will say that two knots K_1 and K_2 are equivalent if there exists an ambient isotopy, allowing a continuous deformation of K_1 into the shape of K_2 without K_1 self-intersecting or breaking. If there does not exist such a continuous deformation or time parameter family of embeddings that evolves K_1 to K_2, then K_1 and K_2 are two distinct types. Many of these distinct types have been tabulated, for example in [16] (see Figure 1). The simplest type of knot is the unknot, or trivial knot. This type of knot is equivalent to the unknotted circle, $x^2 + y^2 = 1, z = 0$.

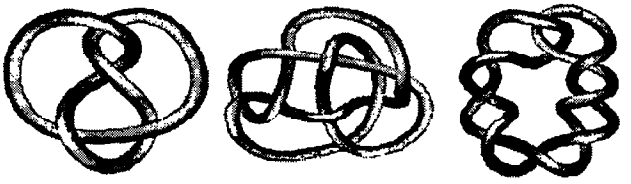

Figure 1: Three distinct knots from Reidemeister's table: $4_1, 9_{33}, 9_2$.

Establishing knot equivalence is a hard problem and no computationally infallible techniques exist for the general case. One method to determine knot equivalence, employs the use of Reidemeister moves. A Reidemeister move is one of 3 rules that changes the crossings of a planar knot projection in a local area while preserving the knot type. If there exists a sequence of Reidemeister moves that deform one knot into another, then the two knots are equivalent. However, a method to determine the correct or minimal sequence of moves to deform one knot to another is still an open problem.

Another method used to determine knot equivalence is to calculate an invariant. An invariant is a unique label associated with each type of knot. If the invariants calculated for two knots are the same, then the knots are of the same type. A polynomial invariant was developed by J.W. Alexander [1]. A polynomial invariant is an algebraic expression that is calculated from the planar knot projection. Different projections of the same knot produce the same polynomial. From the arrangement of crossings of the projection one can determine a set of variables with powers and coefficients unique to a knot type. Alexander's polynomial does not distinguish between all types of knots (i.e., it is not an invariant in a pure sense). Since that time, a number of more powerful polynomial invariants have been discovered, and include the Jones and Kauffman [7] polynomials. However, these do not distinguish all knot types either.

Instead of calculating a polynomial for a knot, one can associate a scalar value (energy) with a knot which can also be used to distinguish among various knot types. Typically, the energy will depend on the geometry of the knot and will vary as the knot undergoes isotopic (i.e., type-preserving) transformations. Therefore, it has been suggested that the invariant be equated to the value of energy of the configuration that minimizes the given functional. A minimal energy configuration is a standard form of the knot and the associated minimal energy is an invariant of knot type [2]. Therefore, the problem of knot classification can be mapped to an optimization problem in which a configuration that minimizes the given energy functional is sought.

A number of energy functions have been proposed for knot in-

0-7803-3707-7/96 ..$4.00

variance. The idea of associating an electrostatic energy to a knot was proposed by Fukuhara [5]. Electrons are placed at equal intervals along a polygonal knot thus causing it to self-repel into a configuration that is a simplification of the original knot. O'Hara and Freedman propose an energy function with electrostatic repulsion and bending energy and prove that with each of their energy functions there exists a finite number of knot types below a given energy value [13]. Buck and Simon show with their "minimal distance" energy function there exists a global energy minimum which is an invariant of knot type. Freedman, He, and Wang [4] give an energy function and prove that energy bounds the average crossing number [4]. Therefore, energy bounds the number of knot types. They also show that the minimum energy configuration of any closed curve is a circle. Kusner and Sullivan experiment with repulsive energy potentials between pairs of line segments or vertices of the knot. They show how energy is invariant under conformal (Mobius) transformations of space [10].

Optimization problems, where a minimum (or maximum) of some scalar cost functional is desired, are often solved by iterative improvement. In this approach, at any stage of the process a change to the current state of the system is proposed (e.g., one of the independent variables is perturbed). If such a change results in an improvement of the evaluated cost function then the change is accepted as the new state. This is iteratively applied until changes no longer give lower cost values. This is a greedy algorithm, allowing only transitions that result in improvement. Gradient descent [14] and conjugate gradient [14] methods are examples of deterministic methods that follow this concept. Deterministic methods such as these do not perform well in the presence of local minima as they will stop at the first minimum found. In practice, one can run such methods a number of times with widely different starting states, but theoretically such procedure is no better than random sampling.

Many of the above optimization techniques were applied to the problem of finding ground states of knots in order to determine knot type. Fukuhara [5] suggested evolving the knot along the gradient of the electrostatic energy potential created by distributing electrons along its length under the constraint that line segments must stay a constant length. His approach is equivalent to using a gradient descent method. Kusner and Sullivan have also applied energy minimization of knots and deformed the knots by the deterministic technique of the conjugate gradient method [10]. Buck and Simon in [2] describe self-repelling knot energies for polygonal knots and suggest evolving the knot along the energy gradient. Simon also suggested an energy function with repulsion along the minimum distance between line segments of a polygonal knot[19]. Here Simon evolves the knot with a nondeterministic descent method by performing small random perturbations of the knot and accepting the move if it lowers the energy.

The above methods use a variety of energy functions but all evolve the knot based on an iterative improvement of the energy. It is generally believed that the various knot energies do have multiple energy minima, although there is no rigorous proof [20]. Employing a greedy optimization technique in such a scenario will likely result in the system getting stuck at the local minimum closest to the initial configuration. Some of the current methods try to introduce *ad hoc* improvements into the scheme in order to escape non-global minima. For example, Kusner and Sullivan [10] keep periodically adding vertices to high energy areas of the knot to assist in untangling. The rationale behind this approach is an intuitive belief that loosening tight areas of the knot will simplify the knot. Similarly, [22] employs noisy function evaluations (i.e., a user-specified fixed percentile of uphill moves are accepted). This simple scheme permits the escaping from some local minima. However, selecting a suitable fraction of such uphill moves is rather difficult–if too few are permitted, the system can only escape very shallow local minima; too many, and the search becomes a purely random sampling of a prohibitively large search space.

In contrast, the research presented here employs an optimization technique that is resilient to getting stuck in local minima through its methodical use of stochastic sampling. In this sampling more promising areas of the search space are dwelled on longer than less promising ones. An important theoretical advantage of simulated annealing over pure stochastic sampling is that the former has global convergence proofs to attest to its ability to find the global minimum. Another, more academic, advantage of the annealing method is that it does not depend upon the energy function being differentiable in order to minimize energy because the gradient of the energy is not used.

Simulated annealing [9] is a non-deterministic technique that operates within an iterative framework similar to methods described earlier: the system evolves from state to state through a series of local perturbations. Configurations that lower the energy of the system are always accepted, resulting in consistent down-hill behavior. However, unlike the case of "greedy" iterative improvement, occasional up-hill moves are also accepted. The probability of accepting such an up-hill move as the new configuration varies during the course of the procedure and is related to a parameter of the system called temperature. Initially, the temperature of the system is high (we say that the system is "melted" and up-hill transitions are accepted often, but as the system "cools down" fewer such transitions are accepted. Finally, when the system "freezes" only down-hill transitions are accepted. Therefore, the system explores the search space in a nearly unconstrained fashion during the early stages of the search slowly narrowing its focus to the more promising areas. In the classical annealing [9] the temperature of the system was lowered logarithmically. A much more efficient algorithm was suggested by Shu [18] who proved that an exponential schedule can be used if the moves are drawn from Lorentzian rather than Gaussian distribution. In either case, such sequence of stochastic perturbations forms a Markov Chain [3], which has a proof of statistical convergence to a global minimum [9]. That is, one can estimate the probability of convergence to a global minimum as a function of the initial conditions.

Applying simulated annealing to minimize knot energies has been independently and concurrently researched by Ligocki [11]. Ligocki's work differs from the work presented here in two significant ways. First, the energy functions evaluated are different. Secondly, the perturbation methods differ, most notably in that Ligocki uses a fixed move size distribution (his geometric perturbation technique allows only one vertex to be moved at a time) requiring a prohibitively slow logarithmic cooling schedule. This forces Ligocki to compromise the theoretical requirements of the conventional annealing and resort to "simulated quenching" which no longer guarantees convergence to the global minimum.

Formally, our problem can be posed as follows. Given an arbitrary configuration of a tangled rope and an energy functional associated with it, determine the globally optimal state and the knot classification thus obtained. First, we show empirically that one of the commonly used functionals has properties that do not make it suitable for techniques employing gradient descent methods (in addition to being non-differentiable). In the solution that we propose, the curve evolves through a series of random local perturbations under the guidance of the simulated annealing algorithm. This allows us to find the ground state unaffected by the numerical or analytical properties of the underlying functional. Two topological methods of randomly perturbing the curve are presented. The method is applied to the un-knot identification problem.

2 Methods

In the proposed method a knot evolves in a self-repelling potential field generated by the functional associated with it. It is guided by

the simulated annealing algorithm in search of the configuration that is globally minimal. To implement simulated annealing, one needs the following four elements. First, a concise description of possible states of the system is required. The second element is a scalar objective function that is applied to the system state. This is the energy function to be minimized. Thirdly, one needs a procedure to introduce random perturbations to the system. Such a procedure, called a "move generator," has to produce valid states and be ergodic (i.e., there has to exist a sequence of perturbations that transforms any two given configuration onto each other). In addition, a move generator has to provide some control over the sizes of the moves so that rapidly converging algorithms can be used. Lastly, an annealing schedule is needed that regulates the changes in system state. The starting (melted) temperature of the system, rate of temperature decrease (cooling), and number of iterations per time step are part of the schedule.

2.1 System Configuration

The system configuration in this method is a polygonal knot in 3 dimensional space described by a list of n vertices, $K : (v_1, v_2, ..., v_n)$. A smooth knot can be replaced by a polygonal knot of the same topological type [2][7]. Therefore, any smooth knot can be input to this system by its polygonalized representation. The system is manipulated in 3D space, representing K in the dimension it is defined within. This is in comparison to knot manipulation methods done only on 2D planar projections.

One can visualize 3D knots by drawing their 2D crossing diagrams or by rendering them in 3D. Unfortunately, both methods tend to result in very cluttered scenes for all but the simplest configurations. Therefore, we will also use a model that is based on the knot's Gauss code (or trip code) [6][15][7] which is a particularly simple transcription of the knot's structure in a plain linear form. In our adaptation, a Gauss model of a knot will be defined to be a circle that is created by enumerating all the vertices of the knot in the order of traversal (see Figure 2 and rows 2-3 on the color plate). Note, that any space curve can be represented this way (even though it may be knotted) and that this type of "space unfolding" has a very different meaning than "knot untangling" which is the subject of this paper. In addition, we will use color to encode selected local properties of knot configurations. For example, we will assign colors based on the potential energy of a link which is useful for identifying "tight spots" (links in tight areas will necessarily have high energies). Alternatively, we will color the links by the strength of the energy gradient (i.e., the magnitude of the self-repulsive force). This method will be useful in investigating local stability issues as links which are in mechanically stable configurations have zero-strength resultant forces acting upon them.

2.2 Energy Functional

Simulated annealing requires a scalar energy function. We have used a knot energy function for polygonal knots proposed by Simon [19] which will be referred to as E_{MD}. The energy is defined on a polygonal knot K of n consecutive edges $e_1, e_2, ..., e_n$. It is a summation of all non-neighboring pairs of edges, defined as follows. Let $len(e_i)$ be the length of edge e_i let $md(e_i, e_j)$ be the minimum distance between non-neighboring edges e_i, e_j, where i, j are not neighboring edges:

$$E_{MD}(K) = \sum_{i,j<=n}^{n} \frac{len(e_i)len(e_j)}{md(e_i, e_j)^2}. \quad (1)$$

The energy function defined by (1) has a number of advantageous mathematical properties. Simon states that for each knot type represented as a polygonal knot, there does exist a configuration that minimizes $E_{MD}(K)$. $E_{MD}(K)$ is scale invariant causing energy to be dependent on the knot's shape, not its size. $E_{MD}(K)$ not only has advantageous theoretical attributes but also has attractive implementation characteristics. The function is defined explicitly for polygonal knots. This is in comparison to some energy functions which are defined for smooth knots and are defined with double integrals for which there is no closed form [2] and so are usually approximated by a double sum. Box energy is also defined for polygonal knots [2] but requires that the knot be sufficiently loose so that the minimum distance between edges be greater than a constant, while $E_{MD}(K)$ does not have this constraint. One (admittedly, fairly academic) drawback of the above functional is that it does not have a continuous derivative and can potentially cause problems with gradient-based approaches. Conveniently, this has no effect on simulated annealing which has a non-analytic nature.

2.3 Perturbation Methods

Another important element of the procedure is the move generator which given the current system configuration will perturb it locally, proposing a change to the system (i.e, it will select a random section of the knot consisting of a random number of vertices and replace it with a different section). The most essential property of a move generator is that it does not change the type of the knot by allowing cross-overs (i.e., passing through itself). One way to perturb a knot is to use purely geometric transformations (e.g., rotate segments randomly). Because such deformations tend to result in crossovers, explicit checks are used in order to reject invalid configurations. This is likely to result in significantly increased computational effort, particularly in tight areas of the knot. An alternative that we adopt in this method employs a physically-based model in order to deform the knot. In this paradigm, the knot is modeled as a physical system whose properties do not allow it to self-penetrate. The perturbation methods presented here constitute a critical technical component of our approach.

Two different perturbation methods are introduced here. During the annealing process the system randomly chooses which perturbation method to apply with equal probabilities. They both treat the knot as a physically based model by associating forces such as electrostatic repulsion and elasticity with the knot. These forces are chosen for their opportunistic qualities. Electrostatic repulsion amongst the edges of the polygonal knot helps push the knot apart, while also maintaining ambient isotopy by not allowing the knot to cross over itself. Spring forces help keep the length of each edge from becoming infinitely long due to the electrostatic repulsive force that is also applied to the edges. It should be noted that our use of electrostatic and elastic forces for the sake of implementing the move generators in no way determines the actual energy associated with the knot which may not have physical interpretation (compare section 2.2).

The ability to control the move size distributions is of critical importance if a computationally tractable implementation is to be achieved [18]. Both methods presented here provide us with such control: the lengths (i.e., number of nodes) of the perturbed segments as well as the time of their physically-based evolution are random variables with Lorentzian distributions.

2.3.1 Physically Based Model

The physically based model applies forces to the knot which are calculated by integration of ordinary differential equations. Two types of forces are used: elastic and electrostatic. Elastic force is implemented with Hook's law: for a pair of particles at positions a and b, spring force between them is given by[21]:

$$f_a = \left[k_s(|l| - r) + k_d \frac{(v_a - v_b) \cdot l)}{|l|}\right] \frac{l}{|l|}, f_b = -f_a \quad (2)$$

where f_a is the force on a, f_b is the force on b, $l = a - b$, r is rest length, k_s is spring constant, k_d is damping constant v_a is the

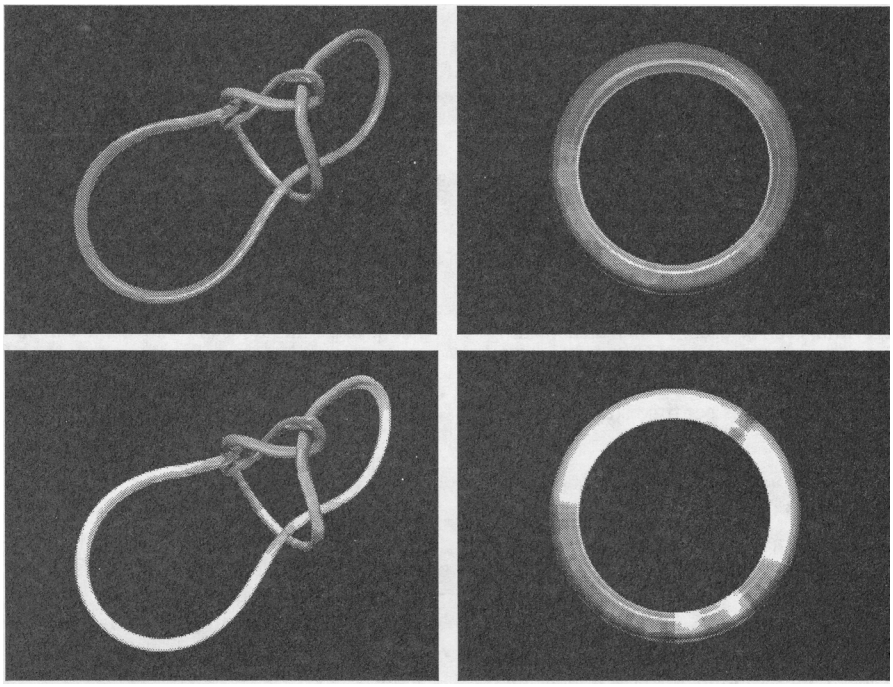

Figure 2: Gauss model of a space curve (right column) is a circle that is created by enumerating all the vertices of the knot (left column) in the order of traversal. Top row is colored (see color plate) by the potential energy (red corresponds to tight, and blue to loose areas) while the bottom row, is colored by the magnitude of the energy gradient (red corresponds to strong, blue to weak force).

velocity of a, v_b is the velocity on b. Electrostatic force is based on Coulomb's law [17] where force is inversely proportional to the distance between two edges. Let a and b be the two edges. Electrostatic force between them is approximated by the equation:

$$f_a = \frac{k(q_a q_b)}{d_{min}} u, f_b = -f_a \tag{3}$$

where f_a is the force on a, f_b is the force on b, q_a is the charge of f_a, q_b is the charge of f_b, d_{min} is the shortest distance between a and b, u is a unit vector pointing from b to a, k is a constant of proportionality.

These forces are applied to the polygonal knot in the following manner. Springs are applied between all neighboring vertices. Electrostatic repulsion is applied between all pairs of edges, except neighboring edges (due to singularities that are present there). Electrostatic repulsion is calculated based on the minimum distance between two line segments in 3D which causes the force to grow infinitely large as two segments approach each other thus preventing crossovers.

In the remainder of this paper, reference to spring force will refer to applying equation (3.1) to all neighboring vertices. And edge repulsion will refer to applying equation (3.2) to all the edge pairs except neighboring edges.

Given the initial configuration of a polygonal knot and the derivative functions which describe the forces acting on the knot, the vertex positions of the knot can be advanced. Integration of the ordinary differential equations was accomplished with a variable step size midpoint Euler method. The Euler method was chosen over Runge-Kutta, which is more accurate, because it is more efficient requiring the evaluation of f only two times per step versus four times with Runge-Kutta. Since hundreds of evaluations may be required for each perturbation, an efficient numeric method was needed.

2.3.2 Charge Drop

The first perturbation method will be referred to as the random charge drop method. A random section of knot is chosen and a random set of point charges are dropped near it. The system is advanced a random number of iterations with variable step size midpoint Euler method only allowing the selected section to move, while the rest of the knot remains fixed. The following forces are applied to the system: elastic, edge repulsion, and electrostatic repulsion between the chosen section and random charges. Line segment repulsion is weighted heavier than random charge repulsion to insure that the force from charge repulsion will not overpower line segment repulsion, resulting in a crossover. This perturbation method pushes a variable size segment of the knot in a random direction while maintaining the knot's topological type (Figure 3).

2.3.3 Node Shift

The second perturbation method is the node shift method which tightens and loosens different parts of the knot. Nodes, or vertices, are shifted along the polygonal knot followed by the application of spring force and edge repulsion forces to the knot. With the newly shifted positions of nodes and the evolution of the system under these forces, this method tightens and loosens parts of the knot. Where nodes have been shifted away, the spring forces will pull these areas together, tightening an area. In the location where nodes have been shifted, spring forces will push apart, loosening this part of the knot (Figure 4). A similar method proposed in [10] adds vertices to a polygonal knot in places where potential energy is high. The node shift method presented in this paper maintains a consistent number of vertices in the knot. Vertices are deleted in some areas and added to other areas which can be viewed as a shift of vertices along the polygonal knot. The number deleted and the number added are equal which maintains a constant problem size making the optimization problem manageable. It also gives consistency to any mapping the user may have defined of attributes to each node.

Nodes are shifted by first selecting vertex positions to delete. A

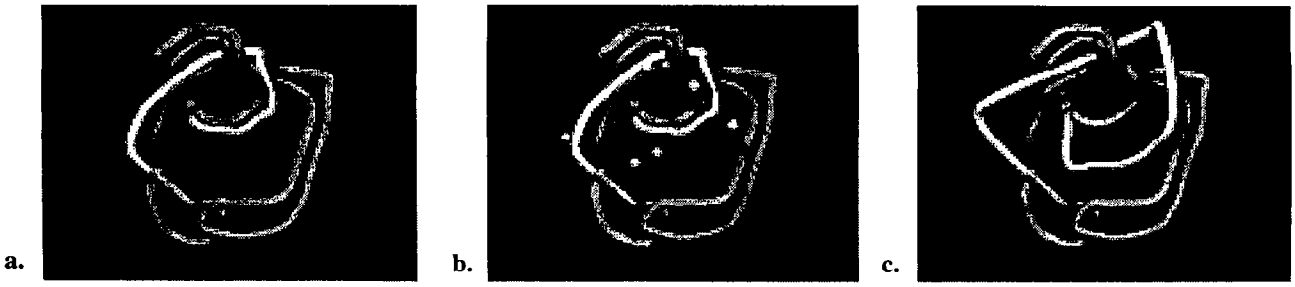

Figure 3: Drop charge perturbation: a. choose random segment, b. drop random charges, c. evolve the system deterministically.

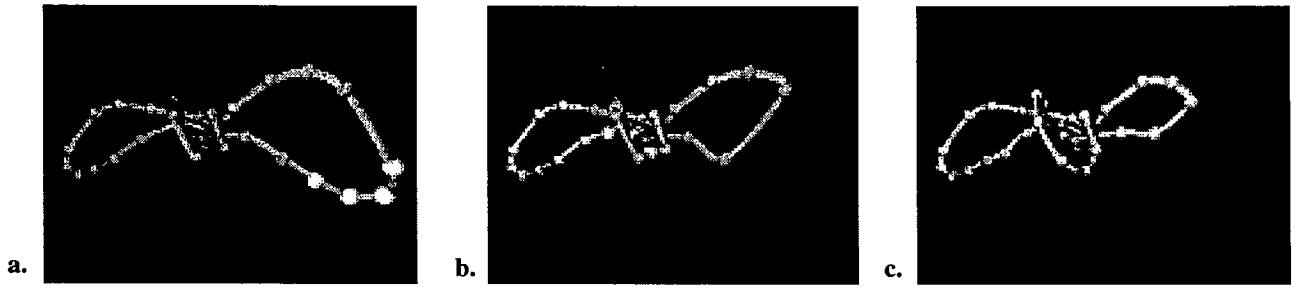

Figure 4: Shift node perturbation: a. nodes are chosen to be removed in a way that will not change the knot type, b. nodes are placed elsewhere on the knot, c. the knot is evolved according to he equations of motion.

position can be deleted if the new knot configuration K' is ambient isotopic to the original knot K. This will be true if the move from K to K' does not allow any part of the knot to pass through itself. This is determined by creating a triangle from the vertices of the selected node and its two neighbors. This triangle represents the smooth path the line segment would travel through in order to move to K' (Figure 5). If any edge intersects this triangle then this movement would create a crossover, possibly changing the knot type; the move would be rejected. However, if no edge intersects this triangle then the node is a candidate for deletion. This triangle is only compared against line segments that are within a threshold distance from it. Also, if the 3 nodes closely approximate a straight line, we assume no line segment will intersect this skinny triangle and delete the vertex without explicit triangle line segment intersection test. A list of nodes to delete are selected by beginning at a start node chosen on the knot then walking along the knot performing triangle tests on consecutive 3 vertices.

Next a set of vertex positions are created. The method walks along the knot creating a new vertex position at the midpoint of edges (see Figure 5). This simple operation cannot affect the knot type.

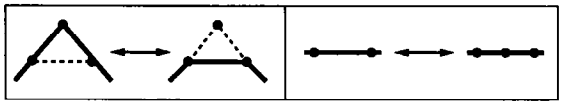

Figure 5: Shifting vertices: remove vertex by triangle move, add vertex.

The list of original vertices and lists of delete and add vertices are properly merged to create a new knot configuration. This configuration is evolved for a random number of iterations with elastic and edge repulsion forces.

2.4 Annealing Schedule

The annealing schedule regulates the temperature and the convergence of the system. The starting temperature should be high so that the system is considered melted. The system is at an effective melting temperature when at least 80% of moves are accepted [8]. From this point the system is slowly cooled taking care that at each temperature the system should reach equilibrium before the temperature is further decreased. In this work, the temperature is decreased when either the system runs max moves or the number of accepted moves exceeds 0.1 x max, where max is equal to a constant times the number of vertices in the knot. Because our moves are Cauchy-distributed, we can use a fairly rapid exponential annealing schedule in which the temperature T decreases by a constant ratio: $T_n = (T_1/T_0)^n T_0$, where T_0 is starting temperature and $0.90 < (T_1/T_0) < 0.99$ [9]. The system is considered frozen, and annealing stops, when the temperature drops below 1.0 degrees or when it is determined that the knot has evolved into a circle, in which case it is unknotted and in a minimal energy state. The latter case is checked by comparing to within a tolerance the knot K's energy with energy calculated for a trivial knot with the same number of vertices as K and in a circular configuration, the minimal energy configuration of an unknot.

3 Results

Two tangled unknots are the subjects of the case studies described below. Both cases were run on an SGI Onyx system configured either with 6 150 MHZ MIPS R4400 processors or 4 200 MHZ MIPS R4400 processors. The problem sizes (as measured by the number of vertices in the model description) were 67 and 139 vertices respectively. For each case, both the deterministic iterative improvement algorithm and the method proposed here were applied to the initial configuration in order to test their performance in their attempts to simplify them.

Case 1 is an unknot represented by 67 vertices. It was simplified

with a deterministic optimizer after 60000 steps going from the initial energy of 1561.85 to the final energy of 116.83, within a threshold value of 0.06 of the global minimum, after 5 hours. Subsequently, the case 1 unknot was subjected to the stochastic optimization method and reached minimal energy configuration within 115 moves which took approximately 2 hours. Temperature started at 150.0 degrees and was decreased exponentially by a constant factor of 0.95. The annealing was considered complete (and the system deemed frozen) when the temperature dropped below 1.0 degrees or when the system's energy E was within $\epsilon = 0.5$ of the global minimum E_{min} of an unknot of 67 vertices. The actual comparison normalized the energy by dividing energy by the number of vertices $vnum$ ($|\frac{E-E_{min}}{vnum}| < \epsilon$). Final energy was 112.84 (as compared to energy of a 67 vertex unknot of 109.96). configuration to the resting state for the proposed method.

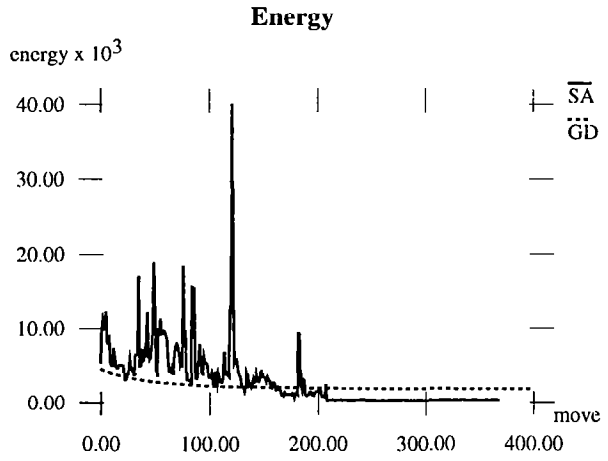

Figure 8: Graph of energy vs. time for Ochiai evolution (case 2, see text).

Case 2 is an unknot represented by 139 vertices whose tangled configuration was described by Ochiai [12] (see top row of the color plate for a stereo pair). The energy of the initial configuration was 4464.47. Gradient descent technique was used to untangle this configuration. Care had to be taken in selecting a suitably small step size, otherwise, the system would attempt energy-increasing moves. Sufficiently small step size would eventually produce a circular configuration. However, the convergence rate of the system was fairly slow: it took 292,000 steps and 107 hours of computation to simplify the knot. Attempts to speed up the process via adaptive step adjustment based on the maximum gradient strength did not produce satisfactory results. Figure 7 (replicated in the two bottom rows of the color plate) shows distribution of the energy gradient strength along the curve for the initial and an intermediate configuration. Simulated annealing reached the minimal energy configuration within tolerance of $\epsilon = 0.5$ after 368 moves. Figure 6 illustrates the knot's start configuration and a number of intermediary configurations during simulated annealing. Figure 8 shows the associated energy graph. Temperature started at 377 degrees, and was allowed to decrease by a factor of 0.95, until temperature reached 1.0 degrees or the energy was within ϵ of the energy of a 139 vertex unknot. The end energy of 235.30 was reached after approximately 48 hours of compute time. Energy of a 139 vertex unknot is 228.52.

4 Conclusions

The energy minimization results showed 2 successful cases. The methods presented here were able to determine the global minimum energy of each case and identify the knots as equivalent to the unknot. From the energy graphs one can see that the knots were allowed to search the solution space and take uphill energy moves. The second case illustrated an unknot configuration that pure gradient descent had difficulty minimizing in an efficient manner whereas simulated annealing successfully found the energy minimum. The result of the second test shows empirically that in this specific case Simon's energy is probably unimodal (i.e., has a single extremum) but that it has a shape of a narrow, nearly-flat, curved hyper-trough that is difficult to follow with gradient descent methods. This conclusion is verified by the bottom row of Figure 7 which depicts the distribution of the gradient strength along the curve in an intermediate configuration arrived at with the iterative improvement method. Nearly zero-strength resultant forces acting on all the links of the knot are indicative of mechanical stability of this configuration which cause slow convergence of the gradient-based methods.

The methods proposed here show promising results as compared to competing methods. Many current methods use gradient descent techniques [2] which will find a minimum but not necessarily the global minimum. Even if all proposed self-repelling potentials are proven to be unimodal our method may produce more rapid convergence than conventional techniques. Some methods use *ad hoc* techniques such as occasionally adding vertices along the knot where energy is high while performing gradient descent [10]. Still other methods do employ occasional uphill energy moves. For instance, the program made available by Wu can be run not only as pure gradient descent but also set to accept occasional uphill energy moves. Some of these methods work well in practice and have faster execution times than the methods developed here, but they lack the the theoretical basis of simulated annealing for finding globally minimal energies and are likely to fail for more complicated cases.

Acknowledgements

Ying-Quing Wu is acknowledged for kindly providing us with a copy of his "ming" program which was used to evaluate the iterative improvement approach. POV-Ray was used for ray-tracing. Dan Sandin is acknowledged for helpful discussion. The first author wishes to acknowledge funding for the Electronic Visualization Laboratory at the University of Illinois at Chicago from the National Science Foundation (NSF), grant CDA-9303433, which includes major support from the Defense Advanced Research Projects Agency.

References

[1] J.W. Alexander. Topological invariants of knots and links. *Trans. Amer. Math. Soc.*, 20:275–306, 1923.

[2] G. Buck and J. Simon. Knots as dynamical systems. *Topology Appl.*, 51, 1993.

[3] G.C. Fox, R.D. Williams, and P.C. Messina. *Parallel Computing Works.* Morgan Kaufmann Publishers, Inc., 1994.

[4] M.H. Freedman, Z.X. He, and Z. Wang. On the energy of knots and unknots. Technical report, Geometry Center Research Report GCG40, University of Minnesota, 1992.

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

[6] C.F. Gauss. *Werke.* Teubner, Leipzig, 1900.

[7] L.H. Kauffman. *Knots and Physics.* World Scientific Publishing, 1993.

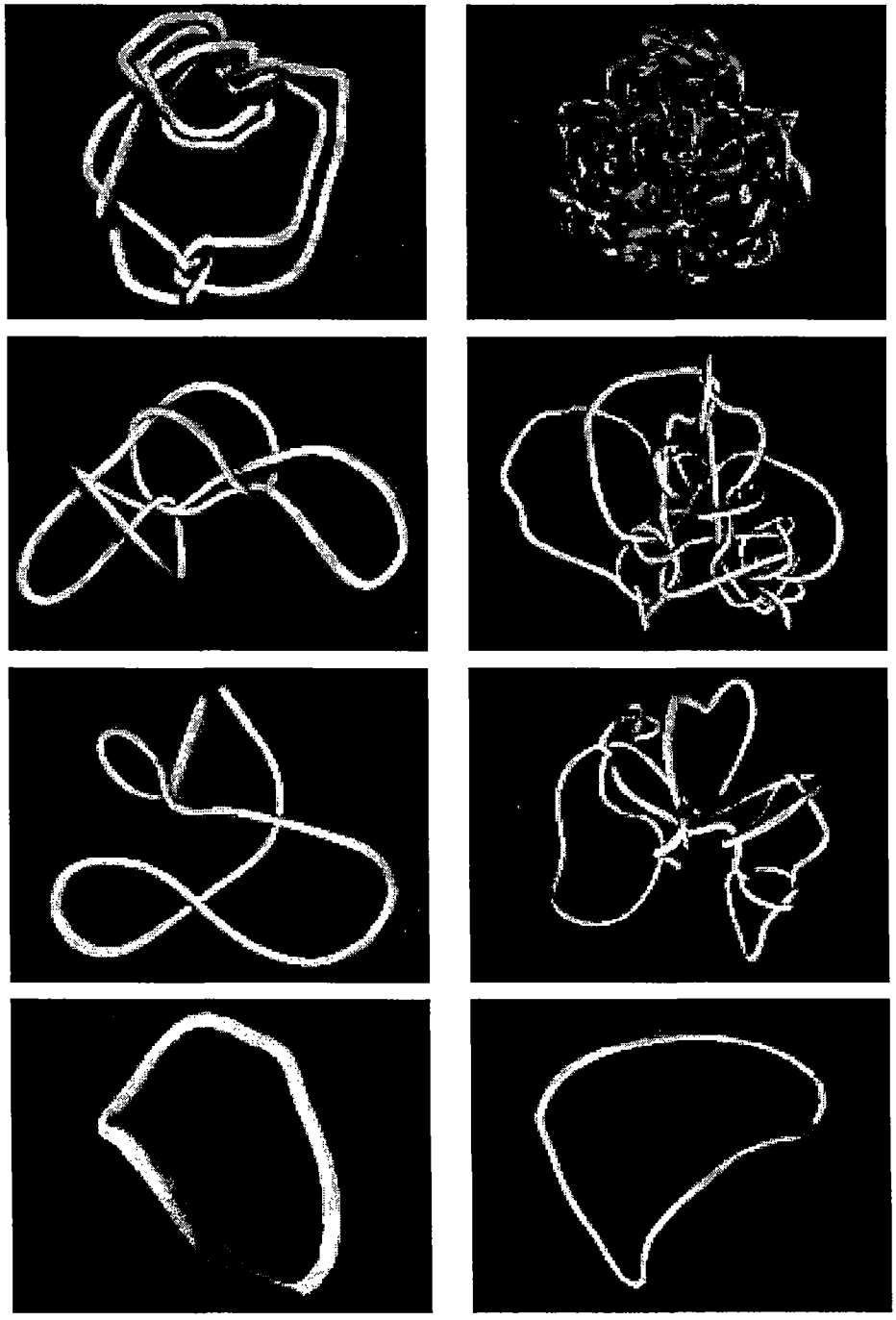

Figure 6: Stochastic evolution of the first (left) and second (right) case studies (see text) from the initial (top), through melted and cooling, to freezing (bottom) stages.

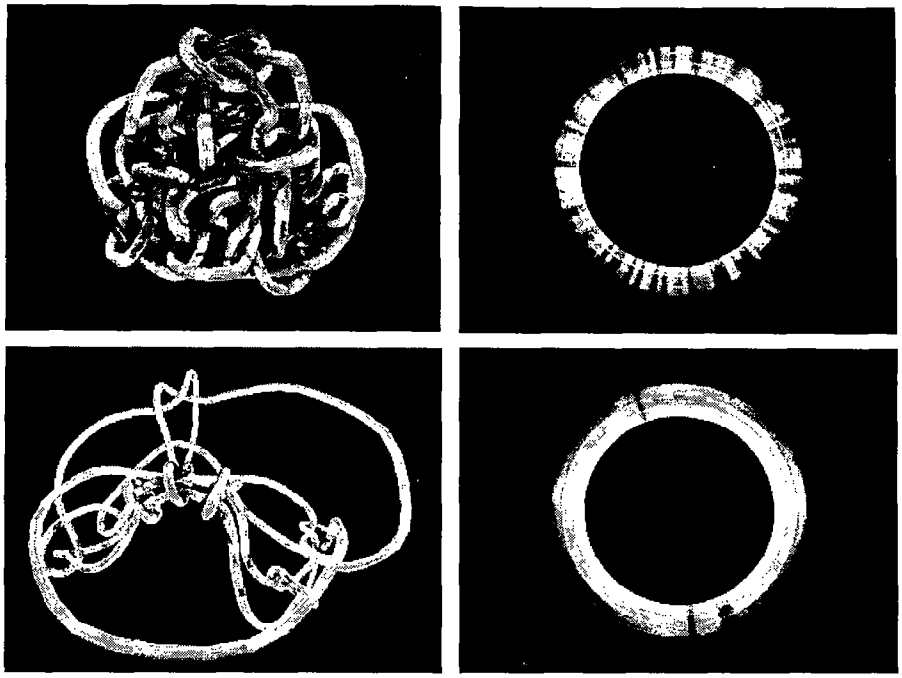

Figure 7: Distribution of the energy gradient magnitude (i.e., strength of the self-repulsive force) along the curve for the initial (top) and an intermediate (bottom) configuration in the iterative improvement approach. The right column shows the configuration on the left after it was unfolded into its Gauss model. Red corresponds to high and blue to low gradient magnitude (see color plate). Nearly zero force for the second configuration indicates a mechanically stable state that gives little guidance to gradient descent methods.

[8] S. Kirkpatrick. Optimization by simulated annealing: Quantitative studies. *Journal of Statistical Physics*, 34:975–986, 1984.

[9] S. Kirkpatrick, C.D. Gelatt, and M.P. Vecchi. Optimization by simulated annealing. *Science*, 220:671–680, 1983.

[10] R.B. Kusner and J.M. Sullivan. Mobius energies for knots and links, surfaces and submanifolds. Technical report, Geometry Center Research Report GCG64, University of Minnesota, 1994.

[11] Terry Ligocki and James Sethian. Recognizing knots using simulated annealing. *Journal of Knot Theory and Its Ramifications*, pages 477–495, 1994.

[12] M. Ochiai. Non-trivial projections of the trivial knot. *S.M.F. Asterisque*, 192:7–9, 1990.

[13] J. O'Hara. Energy of a knot. *Topology*, 30:241–247, 1991.

[14] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling. *Numerical Recipes in C*. Cambridge University Press, 1988.

[15] R.C. Read and P. Rosenstiehl. On the gauss crossing problem. *Colloquia Math. Soc. J. Bolyai, Combinatorics*, 18:843–876, 1976.

[16] K. Reidemeister. *Knotentheorie*. Chelsea Publishing Co., 1948.

[17] H.M. Schey. *Div Grad Curl and All That: An Informal Text on Vector Calculus*. W.W. Norton and Company, Inc., 1992.

[18] H. Shu and R. Hartley. Fast simulated annealing. *Phys. Lett. A.*, 122:157–162, 1987.

[19] J. Simon. Energy functions for polygonal knots. In *Random Knotting and Linking*. World Scientific Publishing, 1994.

[20] J. Simon. Personal correspondence, Jan 1996.

[21] A. Witkin. Particle system dynamics. SIGGRAPH '95 Course Notes No.34: Physically Based Modeling, August 1995.

[22] Ying-Qing Wu. Personal correspondence, Jan 1996.

Ochiai stereo pair:

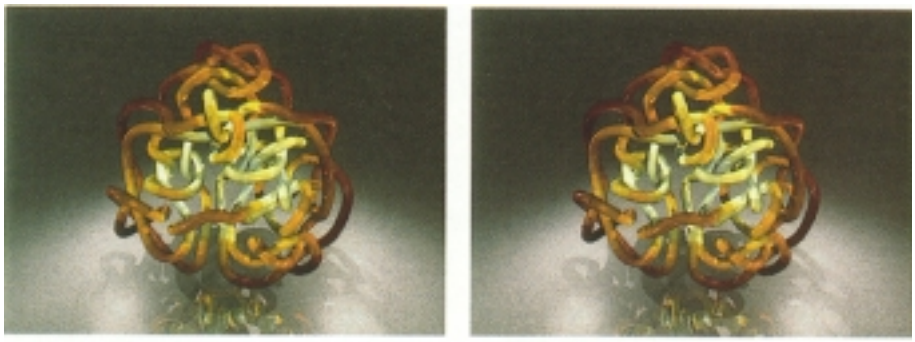

Figure 2:

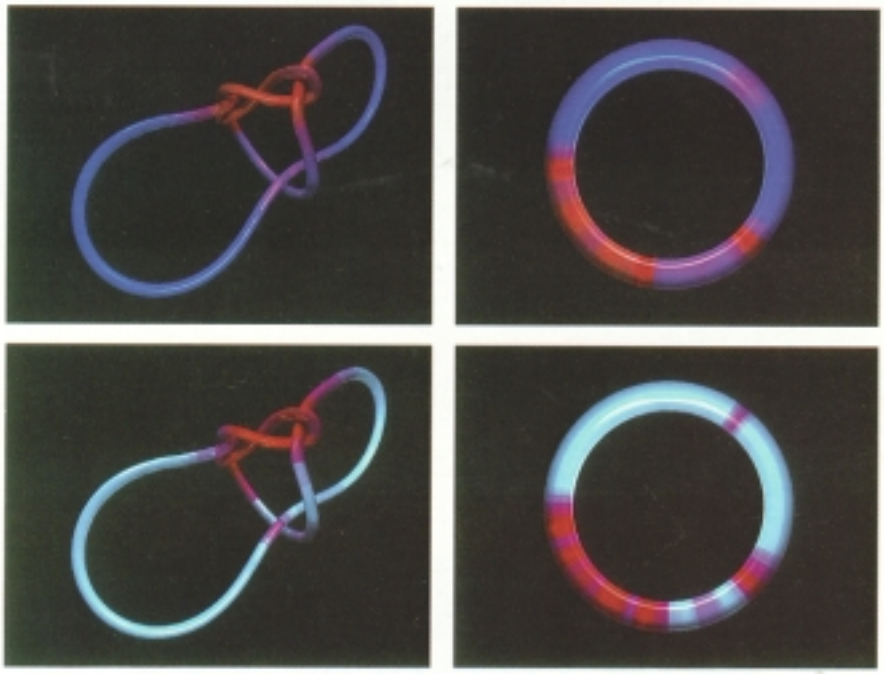

Figure 7: