Evolution of unknotting strategies for knots and braids

Nicholas Jackson[1] and **Colin G. Johnson**[2]

Abstract. This paper explores the problem of *unknotting* closed braids and classical knots in mathematical knot theory. We apply evolutionary computation methods to learn sequences of moves that simplify knot diagrams, and show that this can be effective both when the evolution is carried out for individual knots and when a generic sequence of moves is evolved for a set of knots.

1 INTRODUCTION

1.1 Knots and links

Knot theory is currently one of the richest and most vibrant areas of pure mathematics, having connections not only with other topics in algebraic and geometric topology, but also with many other branches of mathematics, as well as mathematical physics [20] and biochemistry [18].

A full introduction to the study of knots and links is beyond the scope of this article, but a readable introduction may be found in, for example, the book by Cromwell [8], and a more comprehensive but still accessible survey in Rolfsen's classic text [17].
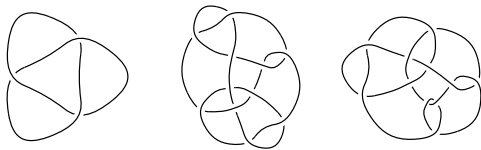

Figure 1. Examples of knots: the trefoil (3_1), Conway's knot ($11n34$) and the Kinoshita–Terasaka knot ($11n42$)

We define a *knot* to be an isotopy class of embeddings $K \colon S^1 \hookrightarrow \mathbb{R}^3$, where $S^1 = \{e^{i\theta} \in \mathbb{C} : 0 \leqslant \theta < 2\pi\}$ denotes the standard unit circle; informally, this is a set of placements of a closed circle in space. A *link* is a knot with more than one circular component, that is, an (isotopy class of an) embedding $L \colon S^1 \sqcup \ldots \sqcup S^1 \hookrightarrow \mathbb{R}^3$. (Alternatively, a knot may be regarded as a link with a single component.)

We generally represent knots and links with diagrams in the plane: projections of the embedded circle(s) where each double intersection point is equipped with crossing information, and we disallow cusps, tangencies or triple-points. Examples may be seen in Figure 1. Identifiers such as 3_1 and 8_{17} refer to the table in Rolfsen's book [17], while identifiers of the form $11a367$ and $11n34$ refer to, respectively, alternating and non-alternating knots in the census of Hoste, Thistlethwaite and Weeks [10].

[1] Mathematics Institute, University of Warwick, Coventry CV4 7AL, UK. Email: Nicholas.Jackson@warwick.ac.uk
[2] School of Computing, University of Kent, Canterbury CT2 7NF, UK. Email: C.G.Johnson@kent.ac.uk

Isotopy of embeddings descends to certain allowable local moves on diagrams which were first studied by Reidemeister [16] and by Alexander and Briggs [2]. These *Reidemeister moves* are depicted in Figure 2. Two knots or links are isotopic if and only if their diagrams are connected by a finite sequence of Reidemeister moves and continuous deformations of the ambient projection plane.
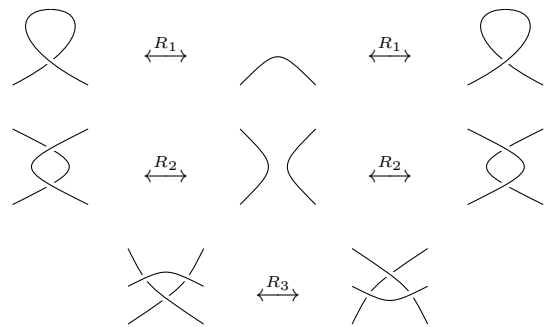

Figure 2. Reidemeister moves

There are a number of different measures of the complexity of a given knot or link K, the best known of which is the *crossing number*: the minimal number of crossings over all possible diagrams for K. Related to this is the *unknotting number $u(K)$*: the minimal number, over all possible diagrams for a knot, of crossings which must be changed in order to obtain a trivial knot. The trefoil in Figure 1 has unknotting number $u(3_1) = 1$: it may be seen to be nontrivially knotted (that is, not isotopic to an unknotted circle) but changing any single crossing results in a diagram which may be transformed (by means of an R_2 move followed by an R_1 move) into an unknotted circle. Less obviously, the other two knots in Figure 1 are also known to have unknotting number 1.

The unknotting number $u(K)$ is a conceptually simple measure of the complexity of a given knot K (broadly speaking, the higher the unknotting number, the more knotted the knot in question) but one which is often not straightforward to calculate. According to the *KnotInfo* database [12], the unknotting number is currently unknown for nine of the 165 prime knots with ten crossings (and also for many knots of higher crossing number); for these nine knots, the unknotting number is is known to be either 2 or 3, due at least in part to work on Heegaard Floer homology by Oszváth and Szabó [15] which ruled out unknotting number 1. Recent work by Borodzik and Friedl [6] has introduced a new invariant which provides a lower bound of 3 for the unknotting number of twenty-five otherwise difficult cases of knots with up to twelve crossings.

In this paper, our main goal is not necessarily to find optimal

bounds on the unknotting numbers of currently unresolved cases, but to explore the possibilities afforded by applying evolutionary computing techniques to pure mathematical problems in group theory and geometric topology, and to try to obtain some qualitative understanding of the search landscape for these problems. The unknotting problem is relatively straightforward to describe and implement, and thus provides a good candidate for a preliminary investigation of this type.

1.2 Braids

The theory of braids was first seriously investigated by Artin [3], and again a full treatment is far beyond the scope of this article, so we will discuss only those aspects essential for what follows. A more comprehensive discussion may be found in either the book by Hansen [9] or the classic monograph by Birman [5].

We define a geometric *braid* on n strings to be a system of n disjoint, embedded arcs $A = \{A_1, \ldots, A_n\}$ in $\mathbb{R}^2 \times [0, 1]$, such that the arc A_i joins the point $P_i = (i, 0, 1)$ to the point $Q_{\tau(i)} = (\tau(i), 0, 0)$, where τ denotes some permutation of the numbers $\{1, \ldots, n\}$, such that each arc A_i intersects the intermediate plane $\mathbb{R}^2 \times \{z\}$ exactly once, for all $0 < z < 1$. Figure 3 shows an example of a 4–string braid.
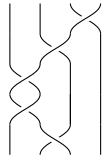

Figure 3.   A 4–string braid

The *elementary braid* σ_i, for $1 \leqslant i \leqslant n-1$, is the n–string braid in which the $(i+1)$st string crosses over the ith string, and no other interactions take place; its inverse σ_i^{-1} is the braid in which the ith string crosses over the $(i+1)$st string (see Figure 4). Any n–string braid β may be represented (although not, in general, uniquely) as a concatenated sequence of elementary braids.

Figure 4.   The elementary braids σ_i and σ_i^{-1}

We consider two n–braids β_1 and β_2 to be equivalent if they are related by an isotopy which keeps their endpoints fixed. In the language of elementary braids, this translates to the following identities:

$$\sigma_i \sigma_j = \sigma_j \sigma_i \qquad \text{for } |i - j| > 1 \qquad (1)$$
$$\sigma_i \sigma_{i+1} \sigma_i = \sigma_{i+1} \sigma_i \sigma_{i+1} \qquad \text{for } 1 \leqslant i \leqslant n - 1 \qquad (2)$$

Geometrically, the first of these corresponds to moving two non-interacting elementary braids past each other, and the second is essentially the third Reidemeister move R_3.

We may define the *braid group* B_n by the following presentation:

$$\left\langle \sigma_1, \ldots, \sigma_{n-1} \middle| \begin{array}{ll} \sigma_i \sigma_j = \sigma_j \sigma_i & |i - j| > 1 \\ \sigma_i \sigma_{i+1} \sigma_i = \sigma_{i+1} \sigma_i \sigma_{i+1} & 1 \leqslant i \leqslant n - 1 \end{array} \right\rangle \quad (3)$$

It can be shown that the group defined by this presentation is isomorphic to the group obtained by imposing the obvious concatenation operation on the (in general, infinite) set of all n–string geometric braids. In this latter group, the identity element is the trivial n–braid (the one with no crossings) and for any braid β, the inverse β^{-1} may be obtained by reflecting β in the horizontal plane $\mathbb{R}^2 \times \{\frac{1}{2}\}$.

There are other, equivalent constructions of the n–string braid group, including one in terms of the fundamental group of a particular configuration space factored by an action of the symmetric group S_n but these will not concern us here.

Given a braid $\beta \in B_n$, we can obtain a link $\widehat{\beta}$ by the *closure* operation depicted in Figure 5, that is, we join each point $P_i = (i, 0, 1)$ to the point $Q_i = (i, 0, 0)$. In fact, Alexander's Theorem [1] (see also Birman [5, Theorem 2.1]) states that any knot or link can be obtained in this way; an explicit algorithm may be found in the paper by Vogel [19]. Note that a closed-braid presentation need not be minimal with respect to the crossing number of the knot. That is, an n–crossing knot might not have a closed-braid presentation with n crossings. Table 1 lists several examples of non-minimal presentations.
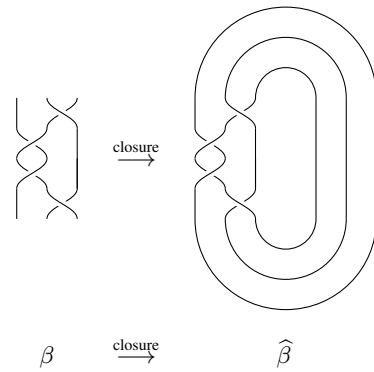
Figure 5.   The closure operation on braids

The following theorem gives explicit conditions for when two different braids yield isotopic knots or links. This result was due originally to Markov, although a full proof was only published some years later by Birman [5, Theorem 2.3] (see also the paper by Rourke and Lambropoulou [11]).

Theorem 1 (Markov [14])**.** *Two braids $\beta_1 \in B_m$ and $\beta_2 \in B_n$ yield closures $\widehat{\beta}_1$ and $\widehat{\beta}_2$ which are isotopic as links if and only if β_1 and β_2 are connected by a finite sequence of moves of type M_1 (conjugation) and M_2 (stabilisation), as depicted in Figure 6.*

2   UNKNOTTING

We now attempt to use evolutionary techniques to devise optimal unknotting strategies for knots and links. The theorems of Alexander and Markov enable us to represent knots and links as words in the standard generators σ_i of the braid group B_n for some n. The

K	Braid word	Strands	Crossings
3_1	σ_1^3	2	3
4_1	$\sigma_1\sigma_2^{-1}\sigma_1\sigma_2^{-1}$	3	4
5_1	σ_1^5	2	5
5_2	$\sigma_1^3\sigma_2\sigma_1^{-1}\sigma_2$	3	6
6_1	$\sigma_1^2\sigma_2\sigma_1^{-1}\sigma_3^{-1}\sigma_2\sigma_3^{-1}$	4	7
6_2	$\sigma_1^3\sigma_2^{-1}\sigma_1\sigma_2^{-1}$	3	6
6_3	$\sigma_1^2\sigma_2^{-1}\sigma_1\sigma_2^{-2}$	3	6
7_1	σ_1^7	2	7
7_2	$\sigma_1^3\sigma_2\sigma_1^{-1}\sigma_2\sigma_3\sigma_2^{-1}\sigma_3$	4	9
7_3	$\sigma_1\sigma_1\sigma_1\sigma_1\sigma_1\sigma_2\sigma_1^{-1}\sigma_2$	3	8
7_4	$\sigma_1^2\sigma_2\sigma_1^{-1}\sigma_2^2\sigma_3\sigma_2^{-1}\sigma_3$	4	9
7_5	$\sigma_1^4\sigma_2\sigma_1^{-1}\sigma_2^2$	3	8
7_6	$\sigma_1^2\sigma_2^{-1}\sigma_1\sigma_3\sigma_2^{-1}\sigma_3$	4	7
7_7	$\sigma_1\sigma_2^{-1}\sigma_1\sigma_2^{-1}\sigma_3\sigma_2^{-1}\sigma_3$	4	7
8_1	$\sigma_1^2\sigma_2\sigma_1^{-1}\sigma_2\sigma_3\sigma_2^{-1}\sigma_4^{-1}\sigma_3\sigma_4^{-1}$	5	10
8_2	$\sigma_1^5\sigma_2^{-1}\sigma_1\sigma_2^{-1}$	3	8
8_3	$\sigma_1^2\sigma_2\sigma_1^{-1}\sigma_3\sigma_2\sigma_3\sigma_4^{-1}\sigma_3\sigma_4^{-1}$	5	10
8_4	$\sigma_1^3\sigma_2^{-1}\sigma_1\sigma_2^{-1}\sigma_3^{-1}\sigma_2\sigma_3^{-1}$	4	9
8_5	$\sigma_1^3\sigma_2^{-1}\sigma_1^3\sigma_2^{-1}$	3	8
8_6	$\sigma_1^4\sigma_2\sigma_1^{-1}\sigma_3^{-1}\sigma_2\sigma_3^{-1}$	4	9
8_7	$\sigma_1^4\sigma_2^{-1}\sigma_1\sigma_2^{-1}\sigma_2^{-1}$	3	8
8_8	$\sigma_1^3\sigma_2\sigma_1^{-1}\sigma_3\sigma_2\sigma_3^{-2}$	4	9
8_9	$\sigma_1^3\sigma_2^{-1}\sigma_1\sigma_2^{-3}$	3	8
8_{10}	$\sigma_1^3\sigma_2^{-1}\sigma_1^2\sigma_2^{-2}$	3	8
8_{11}	$\sigma_1^2\sigma_2\sigma_1^{-1}\sigma_2^2\sigma_3^{-1}\sigma_2\sigma_3^{-1}$	4	9
8_{12}	$\sigma_1\sigma_2^{-1}\sigma_1\sigma_3\sigma_2^{-1}\sigma_4^{-1}\sigma_3\sigma_4^{-1}$	5	8
8_{13}	$\sigma_1^2\sigma_2^{-1}\sigma_1\sigma_2^{-2}\sigma_3^{-1}\sigma_2\sigma_3$	4	9
8_{14}	$\sigma_1^3\sigma_2^{-1}\sigma_1\sigma_2\sigma_3\sigma_2\sigma_3^{-1}$	4	9
8_{15}	$\sigma_1^2\sigma_2^{-1}\sigma_1\sigma_3\sigma_2\sigma_3^2\sigma_3$	4	9
8_{16}	$\sigma_1^2\sigma_2^{-1}\sigma_1^2\sigma_2^{-1}\sigma_1\sigma_2^{-1}$	3	8
8_{17}	$\sigma_1^2\sigma_2^{-1}\sigma_1^2\sigma_2^{-1}\sigma_1\sigma_2^{-2}$	3	8
8_{18}	$\sigma_1\sigma_2^{-1}\sigma_1\sigma_2^{-1}\sigma_1\sigma_2^{-1}\sigma_1\sigma_2^{-1}$	3	8
8_{19}	$\sigma_1^3\sigma_2\sigma_1^3\sigma_2$	3	8
8_{20}	$\sigma_1^3\sigma_2^{-1}\sigma_1^{-3}\sigma_2^{-1}$	3	8
8_{21}	$\sigma_1^3\sigma_2\sigma_1^{-2}\sigma_2^2$	3	8

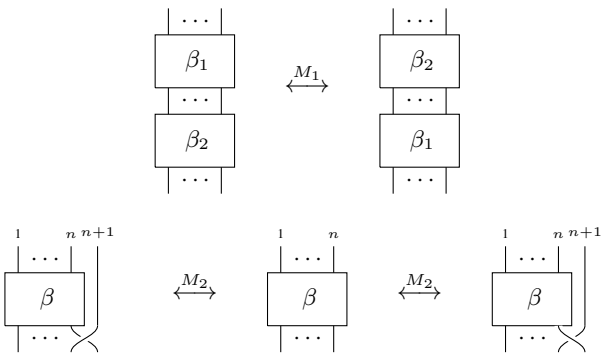**Table 1.** Braid words for knots with up to eight crossings [12]

Figure 6. Markov moves of type M_1 (conjugation) and M_2 (stabilisation)

crossing-change operation is then simply a matter of taking such a word and then replacing a given σ_i with its inverse σ_i^{-1} or vice-versa.

Our goal is, given a knot K represented as the closure of a braid word $w \in B_n$, to evolve a sequence of certain moves which trivialises the knot with the smallest number of crossing-changes, thus obtaining an upper bound on the unknotting number $u(K)$ of K. The allowable moves are those which either leave the isotopy class of the corresponding knot unchanged, or which change the sign of a single crossing, and may be seen in Table 2.

$R_2^\pm$	$\sigma_i^{\pm1}\sigma_i^{\mp1}$	$\longmapsto$	1
$\bar{R}_2^\pm$	1	$\longmapsto$	$\sigma_i^{\pm1}\sigma_i^{\mp1}$
$R_3^\pm$	$\sigma_i^{\pm1}\sigma_{i+1}^{\pm1}\sigma_i^{\pm1}$	$\longmapsto$	$\sigma_{i+1}^{\pm1}\sigma_i^{\pm1}\sigma_{i+1}^{\pm1}$
$\bar{R}_3^\pm$	$\sigma_{i+1}^{\pm1}\sigma_i^{\pm1}\sigma_{i+1}^{\pm1}$	$\longmapsto$	$\sigma_i^{\pm1}\sigma_{i+1}^{\pm1}\sigma_i^{\pm1}$
$M_1^\pm$	$\sigma_i^{\pm1}\alpha$	$\longmapsto$	$\alpha\sigma_i^{\pm1}$
$\bar{M}_1^\pm$	$\alpha\sigma_i^{\pm1}$	$\longmapsto$	$\sigma_i^{\pm1}\alpha$
$M_2^\pm$	$\alpha\sigma_n^{\pm1}$	$\longmapsto$	α
$\bar{M}_2^\pm$	α	$\longmapsto$	$\alpha\sigma_n^{\pm1}$
$S^{\pm\pm}$	$\sigma_i^{\pm1}\sigma_j^{\pm1}$	$\longmapsto$	$\sigma_j^{\pm1}\sigma_i^{\pm1}$ (if $|i-j|>1$)
$\bar{S}^{\pm\pm}$	$\sigma_j^{\pm1}\sigma_i^{\pm1}$	$\longmapsto$	$\sigma_i^{\pm1}\sigma_j^{\pm1}$ (if $|i-j|>1$)
$U^+ = \bar{U}^-$	σ_i	$\longmapsto$	σ_i^{-1}
$U^- = \bar{U}^+$	σ_i^{-1}	$\longmapsto$	σ_i

Table 2. Allowable moves on braid words

In more detail, a move of type R_2^+ cancels a substring of the form $\sigma_i\sigma_i^{-1}$, a move of type R_2^- cancels a substring of the form $\sigma_i^{-1}\sigma_i$, and moves of type $\bar{R}^\pm$ introduce corresponding substrings. These are the Reidemeister moves of type 2, translated into the context of braid words. Similarly, moves of type R_3 perform Reidemeister moves of type 3, moves of type M_1 and M_2 are Markov moves, moves of type S represent non-interacting crossings sliding past each other, and moves of type U change the sign of a single crossing.

3 METHODS

In this paper, we use evolutionary techniques to find sequences of unknotting primitives which are optimal with respect to two subtly different but related problems:

Problem 1. *Given an arbitrary knot K, described as the closure $\widehat{\beta}$ of some braid word $\beta \in B_n$, is there a sequence of moves which reduces β to the trivial word $1 \in B_1$? If so, what is the minimal such sequence with respect to the number of crossing-change operations $U^\pm$, and in what cases does this yield a sharp upper bound for the unknotting number $u(K)$ of K when compared to known values such as those listed in the* KnotInfo *database [12].*

Problem 2. *Given a finite set S of knots, each described as the closure $\widehat{\beta}$ of some braid word $\beta \in B_n$, is there a single, universal sequence of moves which, possibly with repeated applications, trivialises each knot in S?*

The second of these problems is, broadly speaking, the generalisation of the first to more than one reference knot; equivalently, the first problem is the special case of the second where we consider a single knot.

As usual in an evolutionary computation approach, we need to define seven things: how population members are represented; the

parameters; how the population is initialised; how the mutation and crossover operators are defined; and how fitness evaluation and selection happen.

3.1 Representation

Each member of the population consists of a list of moves drawn from those in Table 2. This list can be of any length.

3.2 Parameters

For Problem 1 the population size was 500 and the number of iterations was $4 \times \text{length}(K)^2$, where K is the knot in question and length is the number of crossings. For Problem 2 the population size was 200 and the number of iterations was $4 \times \max(\text{length}(S))^2$, where S is the set of knots. These parameters were determined by informal experimentation. The program was run three times for each knot or set of knots, and the best result is reported.

3.3 Initialisation

Each member of the population is initialised by first choosing a number uniformly at random in the range [1,15] which is the length of the list, and then each slot in the list is filled in uniformly at random from the moves given in Table 2. The moves are selected with replacement; that is, a sequence may include more than one move of a given type.

3.4 Mutation

There are three different mutation operators, which are selected uniformly at random and applied to individuals with the overall mutation probability being 10%. These are:

1. Select a random move from the list and replace it with a move drawn randomly from the moves in Table 2.
2. Choose a random move from the list and delete it (as long as the list of moves contains at least one move).
3. Choose a random position in the list and insert a randomly chosen move from the moves in Table 2.

3.5 Crossover

One-point crossover is applied to all individuals as follows: the two strings are aligned, a position less than or equal to the length of the shortest string is selected at random, and the strings crossed over at that point.

3.6 Fitness evaluation

The members of the population are evaluated by attempting to unknot each of a set of knots, which in the case of Problem 1 will be just a single example, and in the case of Problem 2 will consist of more than one knot. The execution is carried out as follows. Let $M = m_1, m_2, \ldots, m_n$ be the n moves in the list. Let $K = K_0$ be the original knot, and $K_1, K_2, \ldots$ be the sequence of knots generated.

The knot K_0 is analysed for the preconditions for m_1 to be carried out, if they are satisfied then the move is applied, so that $K_1 := m_1(K_0)$; if the preconditions are not satisfied then the knot is unchanged ($K_1 := K_0$). The next step is to attempt to apply m_2 to K_1 by seeing if its preconditions are satisfied, and so on. When the end of M is reached, the list is begun again from the beginning.

Each time the knot is changed as a result of applying the move, the knot is checked to see if there are any crossings remaining. If so, the algorithm terminates, and a positive result returned.

For Problem 1 we apply the sequence M once only, but for Problem 2 we perform repeated applications of M. If the knot hasn't been trivialised by 50 applications of M, we assume that it has become stuck in a repeating loop (which, for the 1 701 936 knots with sixteen or fewer crossings [10], is a valid assumption), terminate the process with a negative result, and move on to the next knot (if any) on the list.

By inspection of Table 2 we see that certain operations (namely, those of type R_2 and M_2) reduce the length of the braid word under investigation, some (types R_3, M_1 and U) don't, and the rest (types $\overline{R}_2$ and $\overline{M}_2$) increase the length of the braid word.

It is known (see, for example, the paper by Coward [7]) that there exist diagrams for the unknot which can only be reduced to the standard (zero-crossing) diagram of the unknot by means of at least one move of type $\overline{R}_2$. That is, at some point during the reduction process, we have to temporarily increase the complexity. In this article however, for simplicity, we will restrict ourselves to sequences of moves which do not increase the length of the braid word. We intend to explore the more general case in later work.

With regard to Problem 1, we are trying to find a sequence of operations which unknot a specific knot with the smallest number of crossing changes. Slightly less importantly, we want to find the simplest possible such unknotting sequence.

For Problem 2, we are trying to find a sequence of operations which (perhaps with repeated applications) unknots as many knots as possible, as efficiently as possible.

Let S denote the set of reference braids, on which each sequence is being tested. (For Problem 1 this will consist of a single braid word.) Let $r_S(M)$ denote the number of braids in S which are fully reduced by (one or more application of) the sequence M. (In practice, we specify an upper threshold of 50 iterations, as described above.) Let $\min_S(M)$ and $\max_S(M)$ denote, respectively, the minimum and maximum number, over all braids in S, of iterations of M required to reduce a (reducible) braid. Let $l(M)$ denote the length of the sequence M. By $c(M)$ we denote the number of crossing-change (type U) operations in M, and by $c_S(M)$ we denote the total number, over all braids in S, of successful crossing-change (type U) operations. That is, $c_S(M)$ gives a measure of the total amount of unknotting actually performed by the sequence M.

In the case of Problem 1, some of the operations in the string M may have no effect on the braid under examination. For example, applying an R_2 move to a braid which at that stage has no $\sigma_i^{\pm 1}\sigma_i^{\mp 1}$ substrings will leave the braid unchanged, and may thus be safely elided from the sequence, resulting in a shorter sequence. Given a sequence M, applied to a braid $\beta \in B_n$, we denote by $l_{\text{opt}}(M)$ the length of the sequence obtained by optimising M in this way with respect to β.

The fitness function should, ideally, seek to minimise the number of crossing changes, maximise (at least when working on Problem 2) the number of knots which can be reduced by a given sequence, minimise the length of the sequence, and minimise the number of repeated applications of the sequence required to reduce those braids in S which are reducible by the operations under consideration.

With those criteria in mind, we define the fitness function for Problem 1 to be

$$f_1(M) = 1 + \frac{10000 r_S(M)}{l_{\text{opt}}(M) + c_S(M)^3 + 1}$$

and that for Problem 2 to be

$$f_2(M) = 1 + \frac{r_S(M)^2}{1 + \max_S(M) + l(M)}.$$

Since both $f_1(M)$ and $f_2(M)$ depend only on the set of braids under consideration, which doesn't change between generations, we can optimise the simulations by caching the fitness values for a given string M, rather than recalculating it each time.

3.7 Selection

Using an approach similar to the Stochastic Universal Sampling Algorithm [4], in each generation, we rank the candidate sequences in order of their normalised fitness $\bar{f}(M)$: the fitness $f(M)$ of the sequence M divided by the mean fitness over the whole population. The integer part of $\bar{f}(M)$ gives the number of copies contributed to the next generation, while the fractional part gives the probability of an additional copy. So, a sequence M with a normalised fitness $\bar{f}(M) = 1.72$ contributes one copy to the following generation, plus a 72% chance of a second copy.

3.8 Implementation

The source code for the implementation is available from the authors on request.

4 RESULTS

4.1 The single unknotting problem

Table 1 lists braid words for all knots with up to eight crossings, and Table 3 lists unknotting sequences for those knots, generated by a Perl program implementing Problem 1. The sequences are not necessarily unique (and in many cases will not be), nor are they guaranteed to be optimal; however we observe that for 21 of the 35 knots with eight or fewer crossings, our program has correctly calculated the unknotting number.

Figure 7 shows the braid $\sigma_1 \sigma_2^{-1} \sigma_1 \sigma_2^{-1}$ (whose closure is isotopic to the figure-eight knot 4_1) being reduced by the sequence $U M_1^2 R_3 R_2 M_2^2$.
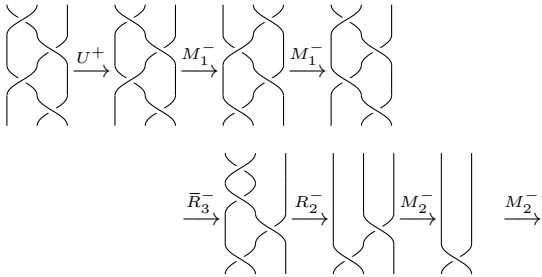

Figure 7. Reduction of the figure-eight knot

The knots 3_1, 5_1 and 7_1 are worth examining a little closer: these are the *torus knots* of type $(2, 2n+1)$, with braid presentation $\sigma_1^{2n+1} \in B_2$ and unknotting number $u(K) = n$. The unknotting

sequences obtained for these knots have a very similar form, namely $(U R_2)^{n-1} M_2$. More generally, given two positive, coprime integers p and q, the *torus knot* of type (p, q) is the knot which can be drawn on the surface of a standard, unknotted torus so that the strands wind p times round the torus in the longitudinal direction, and q times in the meridional direction. The torus knot $T_{p,q}$ of type (p, q) has unknotting number $u(T_{p,q}) = \frac{1}{2}(p-1)(q-1)$. See Rolfsen [17, Section 3.C] or Cromwell [8, Section 1.5] for further details on torus knots.

K	Unknotting sequence M	$c(M)$	$u(K)$
3_1	$U R_2 M_2$	1	1
4_1	$U M_1^2 R_3 R_2 M_2^2$	1	1
5_1	$(U R_2)^2 M_2$	2	2
5_2	$U R_2 M_1^2 R_3 R_2 M_2^2$	1	1
6_1	$U R_2 U M_2 M_1^2 R_3 R_2 M_2^2$	2	1
6_2	$U R_2 U M_1^2 R_3 R_2 M_2^2$	2	1
6_3	$U R_2 U M_2 R_2 M_2$	2	1
7_1	$(U R_2)^3 M_2$	3	3
7_2	$U R_2 M_1 (M_1 U)^2 (M_1 R_3 R_2 M_2)^2 M_2$	3	1
7_3	$(U R_2 M_1)^2 R_3 R_2 M_2^2$	2	2
7_4	$U R_2 M_2 U R_2 M_1^2 R_3 R_2 M_2^2$	2	2
7_5	$(U R_2)^2 M_2 U R_2 M_2$	3	2
7_6	$U R_2 M_2 U M_1^2 R_3 R_2 M_2$	2	1
7_7	$M_1 U M_1^4 (R_3 R_2 M_2)^2 M_2$	1	1
8_1	$(U R_2 M_2)^2 U M_1^2 R_3 R_2 M_2^2$	3	1
8_2	$U R_2 M_1^4 U M_1^2 R_3 M_1 R_2^2 M_2^2$	2	2
8_3	$U R_2 M_2 M_1^2 U M_1^2 R_3 R_2 M_1 M_2 R_3 R_2 M_2^2$	2	2
8_4	$M_1 U R_2 M_1 U M_1^3 (R_3 R_2 M_2)^2 M_2$	2	2
8_5	$U M_1 U R_2 M_1^2 R_3 R_2^2 M_2^2$	2	2
8_6	$(U R_2)^2 M_2 U M_1^2 R_3 R_2 M_2^2$	3	2
8_7	$M_1 U R_2 M_1 U M_1^2 R_3^2 (R_2 M_2)^2$	2	1
8_8	$U R_2 M_1^5 U R_2 M_1^2 R_3 M_1 R_2 M_2^3$	2	2
8_9	$M_1^3 U M_1^2 R_3^3 M_2 R_2^3 M_2$	1	1
8_{10}	$U R_2 U M_1^3 (R_3 R_2)^2 M_2^2$	2	2
8_{11}	$U R_2 M_2 M_1^2 U M_1^2 R_3 M_1 R_2^2 M_2^2$	2	1
8_{12}	$M_1 U M_1^5 U M_1 R_3 M_1 U M_2 R_2 M_2 R_3 R_2 M_2^2$	3	2
8_{13}	$U R_2 M_2 U R_2 M_1^2 R_3 R_2 M_2^2$	2	1
8_{14}	$U R_2 M_1 U M_1^2 U M_1^2 (R_3 R_2 M_2)^2 M_2$	3	1
8_{15}	$U R_2 M_2 M_1^3 U R_2 M_1 R_3 R_2 M_2^2$	2	2
8_{16}	$U R_2 U M_1 U (R_2 M_2)^2$	3	2
8_{17}	$M_1^2 U M_1^3 R_3 M_1 R_2 (R_3 R_2)^2 M_2^2$	1	1
8_{18}	$U M_1^2 R_3 R_2 M_1 U (R_2 M_2)^2$	2	2
8_{19}	$U R_2 M_1 R_3 U M_1 U (R_2 M_2)^2$	3	3
8_{20}	$M_1^5 U R_2 M_1^2 R_3 R_2^2 M_2^2$	1	1
8_{21}	$U R_2 M_1^3 (R_3 R_2)^2 M_2^2$	1	1

Table 3. Unknotting sequences M for single knots K, using the braid words from Table 1, comparing the number $c(M)$ of crossing changes performed by the sequence M, and the crossing number $u(K)$ of the knot K

4.2 The multiple unknotting problem

A simulation of Problem 2, again implemented in Perl, obtains universal or near-universal unknotting sequences, some examples of which are listed in Table 4. Some of the more complex braids (those corresponding to knots with minimal crossing number 9 or higher) were unreducible by any scheme found by our program, because we restricted ourselves to operations which don't increase the length of the braid word. As noted earlier (see the paper by Coward [7] for details), sometimes we need to perform a move of type $\bar{R}_2^{\pm}$ to introduce two additional crossings during the reduction scheme.

S	M	$\max_S(M)$	$r_S(M)$	$|S|$
$3_1\text{–}4_1$	$UM_1^2R_3R_2M_2^2$	1	2	2
$3_1\text{–}4_1$	$UM_1^2R_3R_2M_2$	2	2	2
$3_1\text{–}4_1$	$UR_2R_3M_2M_1^2$	3	2	2
$3_1\text{–}5_2$	$M_1UR_3R_2M_1M_2^2U$	2	4	4
$3_1\text{–}5_2$	$UR_2M_1^2R_3M_2R_2M_2$	2	4	4
$3_1\text{–}5_2$	$UR_3R_2M_1^3R_3M_2^2$	2	4	4
$3_1\text{–}6_3$	$M_1R_2UM_1^2R_3R_2M_2$	3	7	7
$3_1\text{–}6_3$	$M_1UM_1^2R_3R_2M_2$	3	7	7
$3_1\text{–}7_7$	$M_2UR_2M_2M_1UR_3$ $M_1^2M_2$	4	14	14
$3_1\text{–}7_7$	$M_1R_3USM_1R_2$ M_2M_1	5	14	14
$3_1\text{–}7_7$	$UR_3R_2SM_2M_1^2M_2$	6	14	14
$3_1\text{–}8_{21}$	$M_1UR_3M_1^2R_3M_2$ $SR_3R_2M_2^3$	5	35	35
$3_1\text{–}8_{21}$	$M_1UM_1R_3R_2M_2^2$ R_3M_1	7	35	35
$3_1\text{–}9_{49}$	$UR_2R_3M_1^2UM_2$ $R_3M_1M_2M_1R_3$	6	74	84
$3_1\text{–}9_{49}$	$R_3R_2UM_1^2M_2^2$	10	73	84

Table 4. Universal or near-universal unknotting sequences for multiple knots

5 CONCLUSIONS AND FUTURE WORK

In this paper we have seen how an unknotting algorithm can be evolved based on a number of primitive moves. There are a number of future directions for research in the area of applying evolutionary algorithms and other machine learning techniques to mathematical problems in knot theory and related areas.

Initially, there are some basic extensions to the work described in this paper. For example, rather than focusing on unknotting we could use similar techniques to address the related problem of knot equivalence. Furthermore, there are similar problems in other areas of mathematics (for example, graph theory and group theory) so this could be extended to those.

More interestingly, there is the question of counterexample search. There are a number of conjectures in this area where there are some measures that could be used to ascertain how close a particular example is to being a counterexample to the conjecture. Using these measures, experiments could be done on exploring the space of knots to find a counterexample; some preliminary work along these lines has been done by Mahrwald [13].

Another approach would be to take a data mining approach to certain mathematical problems. For example, we could generate a large database of knots and then apply classification techniques to distinguish different classes of knot, or applying clustering techniques to group knots according to some metric.

An examination of the results from this classification might give new insights into the underlying structure of the space of knots. A related topic is using genetic programming to evolve *invariants*, that is, functions that distinguish between different knots by processing their diagrams.

REFERENCES

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

[2] J. W. Alexander and G. B. Briggs, 'On types of knotted curves', *Ann. of Math. (2)*, **28**(1-4), 562–586, (1926/27).

[3] Emil Artin, 'Theorie der Zöpfe', *Abh. Math. Sem. Univ. Hamburg*, **4**, 47–72, (1925).

[4] James E. Baker, 'Reducing bias and inefficiency in the selection algorithm', in *Proceedings of the Second International Conference on Genetic Algorithms and their Applications*, ed., John J. Grefenstette, pp. 14–21, Hillsdale, New Jersey, (1987). L. Erlbaum Associates.

[5] Joan S. Birman, *Braids, links, and mapping class groups*, volume 82 of *Annals of Mathematics Studies*, Princeton University Press, Princeton, N.J., 1974.

[6] Maciej Borodzik and Stefan Friedl. The unknotting number and classical invariants I. arXiv preprint 1203:3225, 2012.

[7] Alexander Coward, 'Ordering the Reidemeister moves of a classical knot', *Algebr. Geom. Topol.*, **6**, 659–671, (2006).

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

[9] Vagn Lundsgaard Hansen, *Braids and coverings: selected topics*, volume 18 of *London Mathematical Society Student Texts*, Cambridge University Press, Cambridge, 1989. With appendices by Lars Gæde and Hugh R. Morton.

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

[11] Sofia Lambropoulou and Colin P. Rourke, 'Markov's theorem in 3–manifolds', *Topology Appl.*, **78**(1-2), 95–122, (1997). Special issue on braid groups and related topics (Jerusalem, 1995).

[12] Charles Livingston and Jae Choon Cha. KnotInfo. http://www.indiana.edu/~knotinfo/.

[13] Valentin Mahrwald, *Search-based Approaches to Knot Equivalence*, MMath dissertation, Department of Computer Science, University of York, May 2008.

[14] A. A. Markov, 'Über die freie Aquivalenz geschlossener Zöpfe', *Rec. Math. Mosc.*, **1**, 73–78, (1935).

[15] Peter Ozsváth and Zoltán Szabó. Knots with unknotting number one and Heegaard Floer homology. arXiv preprint math/0401426, 2004.

[16] Kurt Reidemeister, 'Elementare begründung der knotentheorie', *Abh. Math. Sem. Univ. Hamburg*, **5**, 24–32, (1926).

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

[18] De Witt Sumners, 'Lifting the curtain: using topology to probe the hidden action of enzymes', *Notices Amer. Math. Soc.*, **42**(5), 528–537, (1995).

[19] Pierre Vogel, 'Representation of links by braids: a new algorithm', *Comment. Math. Helv.*, **65**(1), 104–113, (1990).

[20] Edward Witten, 'Quantum field theory and the Jones polynomial', *Comm. Math. Phys.*, **121**(3), 351–399, (1989).