

involve

a journal of mathematics

The linking-unlinking game

Adam Giambrone and Jake Murphy



The linking-unlinking game

Adam Giambrone and Jake Murphy

(Communicated by Kenneth S. Berenhaut)

Combinatorial two-player games have recently been applied to knot theory. Examples of this include the knotting-unknotting game and the region unknotting game, both of which are played on knot shadows. These are turn-based games played by two players, where each player has a separate goal to achieve in order to win the game. In this paper, we introduce the linking-unlinking game which is played on two-component link shadows. We then present winning strategies for the linking-unlinking game played on all shadows of two-component rational tangle closures and played on a large family of general two-component link shadows.

1. Introduction

Recently, a number of researchers have applied game theory to knot theory in the form of combinatorial games played on knot diagrams. Examples of such games include twist untangle [Ganzell et al. 2014], the knotting-unknotting game [Henrich et al. 2011; Johnson 2011], and the region unknotting game [Brown et al. 2017]. The game twist untangle is played between two people on a nontrivial diagram of the unknot formed by iteratively twisting the unknotted circle. Players take turns using either an R1 move or an R2 move (see Figure 6) to decrease the number of crossings and simplify the diagram. The winner is the player that reduces the diagram to the unknotted circle. For more details, see [Ganzell et al. 2014]. The knotting-unknotting game is played on a *shadow* of a knot (see Definition 2.3). Players take turns *resolving* a crossing (see Definition 2.4) until all crossings are resolved and a knot diagram is formed. One player, the unknotter, wins if the resulting knot diagram is unknotted (represents the trivial knot), while the other player, the knotter, wins if the resulting knot diagram is knotted (represents a nontrivial knot). For more details, see [Henrich et al. 2011; Johnson 2011]. The region unknotting game is similar to the knotting-unknotting game in that the goals of the knotter and unknotter remain the same. The key difference is that play now

MSC2010: 57M25, 91A46.

Keywords: knot, knot diagram, link, link diagram, linking-unlinking game, pseudodiagram, rational link, rational tangle, splittable, two-player game, unsplittable, winning strategy.

consists of resolving the set of crossings that are incident to a face of the knot shadow (or changing the crossing type of a crossing if it has already been resolved). For more details, see [Brown et al. 2017].

In this paper, we will adapt the knotting-unknotting game to be played on two-component link shadows. We call this new game the *linking-unlinking game*. Here, players still take turns resolving crossings. One player, the *unlinker*, wins if the resulting two-component link diagram is *splittable* (is equivalent to a two-component link diagram where the components are separated from each other), while the other player, the *linker*, wins if the resulting two-component link diagram is *unsplittable* (is not splittable).

Our main goal in this paper is to find winning strategies for playing the linking-unlinking game on families of two-component link shadows. The first family of link shadows we explore consists of the shadows of the two-component links that arise as a closure of the rational tangle $(a_1, \dots, a_n)$. Roughly speaking, a rational tangle is formed by taking two vertical parallel strands and alternating between applying twists to the bottom two endpoints of the strands and applying twists to the right two endpoints of the strands. For an example of the construction of a rational tangle, see Figure 16. For the precise definition of rational tangle, see Definition 2.14. Our first three main results are combined into the following theorem.

Theorem 1.1. *Suppose we have a shadow of a rational two-component link coming from a closure of the rational tangle $(a_1, \dots, a_n)$:*

- (1) *If $a_{2k+1} = 0$ for all k , then the unlinker wins.*
- (2) *If either*
 - (a) *$a_{2k+1} \neq 0$ for at least one k and all of the a_i are even,*
 - (b) *$n = 2$ and both a_1 and a_2 are odd, or*
 - (c) *$n \geq 3$, both a_1 and a_n are odd, and all other a_i are even,*

then the second player has a winning strategy (regardless of their role).

To extend the results above to all shadows of two-component rational tangle closures, we utilize a decomposition of the syllables of the tangle word $(a_1, \dots, a_n)$ into syllables consisting of self-intersections and strings of syllables consisting of non-self-intersections (see Definition 2.18 and Proposition 2.22). By combining this decomposition with the proof of Theorem 1.1, we are able to prove the following result.

Theorem 1.2. *Suppose we have a shadow of a rational two-component link coming from a closure of the rational tangle $(a_1, \dots, a_n)$. Furthermore, assume $a_{2k+1} \neq 0$ for some k :*

- (1) *If the tangle word contains an even number of self-intersections, then the second player has a winning strategy (regardless of their role).*

- (2) *If the tangle word contains an odd number of self-intersections, then the first player has a winning strategy (regardless of their role).*

Finally, we conclude the paper by expanding our focus to general two-component link shadows, using a decomposition of the crossings of a link shadow into self-intersections and non-self-intersections (see Definition 2.11) and using linking number arguments (see Definitions 2.9, 2.10, and 2.12) to prove the following result.

Theorem 1.3. *Suppose we have a shadow of a general two-component link:*

- (1) *If the shadow contains zero non-self-intersections, then the unlinker wins.*
- (2) *If the shadow contains a nonzero number of non-self-intersections and an even number of self-intersections, then the linker has a winning strategy when playing second.*
- (3) *If the shadow contains a nonzero number of non-self-intersections and an odd number of self-intersections, then the linker has a winning strategy when playing first.*

The remainder of this paper is organized as follows. In Section 2, we present background from knot theory, defining terminology and providing results that will be used later. Specifically, we begin by introducing knots and links, various types of knot and link projections, and notions of equivalence for these objects in Section 2A. In Section 2B, we define splittable and unsplittable link diagrams, self-intersections and non-self-intersections for link diagrams, and the linking number of a two-component link diagram. In Section 2C, we define rational tangles and rational link diagrams. In Section 2D, we define self-intersections and non-self-intersections for rational tangles, determine exactly when a rational tangle will close to form a two-component link diagram, and provide a decomposition theorem for rational tangle words.

In Section 3, we define the linking-unlinking game and present winning strategies for playing the game on various two-component link shadows. Specifically, we define the linking-unlinking game and present two key player strategies in Section 3A, we present winning strategies for all shadows of two-component rational tangle closures in Section 3B, and we present winning strategies for large families of general two-component link shadows in Section 3C.

2. Definitions and background

To begin, we introduce some basic ideas from knot theory. Many of the definitions that follow can be found in introductory knot theory textbooks such as [Johnson and Henrich 2017; Adams 1994].

2A. Knots, knot projections, and knot equivalence. A mathematical knot is much like the everyday knot we tie in our shoelaces. The key difference is that mathematical knots form closed loops.



Figure 1. Some examples of knots.

Definition 2.1. A *knot* is a piecewise-linear simple closed curve in three-dimensional space $\mathbb{R}^3$. Equivalently, a *knot* is a smooth embedding of a circle into $\mathbb{R}^3$.

Returning to our shoelace analogy, if we fused the loose ends of a knotted shoelace together, we would have a model of a mathematical knot. Figure 1 provides some examples of knots. Note that, if we consider the piecewise-linear definition of a knot, we can assume that knots are made up of a very large number of line segments so that the strands of the knot appear smooth. We now shift our attention to links, which consist of knots.

Definition 2.2. A *link* is a collection of one or more knots (that can be, but do not have to be, interlinked). An *n-component link* is a collection of n knots.

Note that a knot is a 1-component link, which means the study of links includes the study of knots. For the majority of this paper, we will focus on two-component links. See Figure 2 for some examples of links with multiple components. To simplify the study of knotted loops in three dimensions, we will carefully project our links to two dimensions.

Definition 2.3. A *link shadow* is a projection of a link onto the plane $\mathbb{R}^2$ so that all crossings are transverse double crossings.

We can think of a link shadow as being formed by carefully shining a flashlight on a link and viewing the shadow it casts. Figure 3 provides both an example and a nonexample of a link shadow.

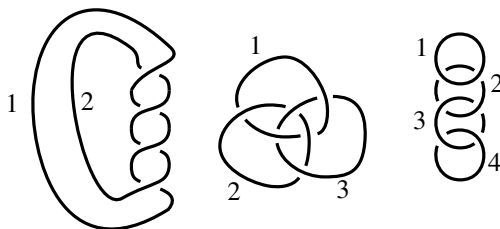


Figure 2. A two-component link (left), a 3-component link (middle), and a 4-component link (right). Note that the components of each link have been numbered.

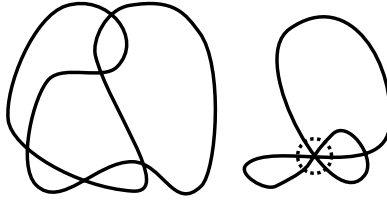


Figure 3. The figure on the left is a valid link shadow. The figure on the right is not, as the circled crossing has three strands meeting at the crossing rather than two.

Notice that link shadows only tell us where crossings occur, not which strand is above the other at each crossing. This means that multiple links can have the same link shadow. To indicate a specific link, we need to include more information.

Definition 2.4. A *resolved crossing* is a crossing in a link projection that has been *resolved* so that the *overstrand* and the *understrand* are distinguishable. This is usually depicted by adding two gaps to the understrand. A crossing that has not been resolved is called an *unresolved crossing*.

To resolve an unresolved crossing, we choose which strand becomes the overstrand. There are two possible resolutions for each crossing, as shown in Figure 4. We now define the results of resolving a subset of crossings of a link shadow.

Definition 2.5. A *link pseudodiagram*, as defined in [Hanaki 2010], is a link projection where an arbitrary number of crossings have been resolved. From this perspective, a link shadow is a link pseudodiagram where no crossings have been resolved and a *link diagram* is a link pseudodiagram where every crossing has been resolved.

For an example of a link pseudodiagram, see Figure 5. Given the family of all links and the family of all link diagrams, we will now discuss notions of equivalence for each of these families.

Definition 2.6. If we can manipulate three-dimensional space $\mathbb{R}^3$ to deform one link L_1 into another link L_2 , then L_1 and L_2 are called *equivalent*.

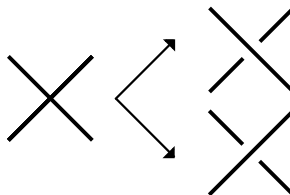


Figure 4. An unresolved crossing of a link projection (left) and the two possible resolutions for this crossing (right).

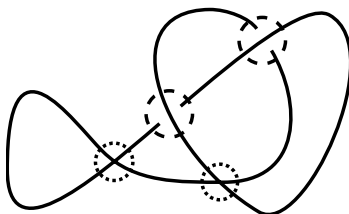


Figure 5. A link pseudodiagram where the upper crossings circled with dashes are resolved and the lower crossings circled with dots are unresolved.

Link equivalence allows the collection of all links to be partitioned into equivalence classes. To determine equivalence on a diagrammatic level, we use *Reidemeister moves* (depicted in Figure 6) and *planar isotopies* (depicted in Figure 7). In this paper, we will focus almost exclusively on the R2 move. Both Reidemeister moves and planar isotopies are local moves, meaning they occur within a fixed region of the plane (so the diagram outside of this region remains unchanged and is, therefore, omitted from Figures 6 and 7). A planar isotopy can be thought of as stretching and bending a single strand of a link diagram without affecting the crossing structure of the diagram. An example of a planar isotopy is shown in Figure 8.

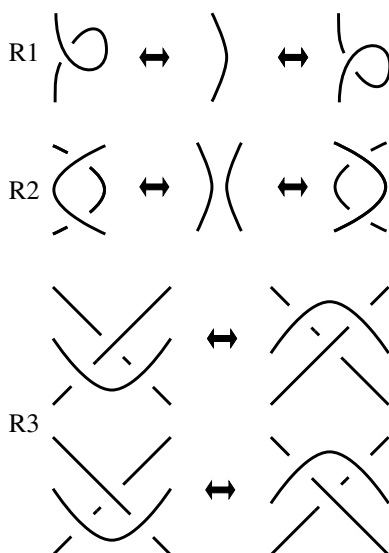


Figure 6. The three Reidemeister moves: R1 moves add or remove a loop, R2 moves overlay one strand on top of a nearby strand (or the reverse process), and R3 moves slide a strand over a crossing. In this paper, we will focus almost exclusively on the R2 move.

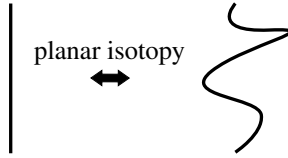


Figure 7. A general planar isotopy.

Definition 2.7. If there is a finite sequence of Reidemeister moves and planar isotopies that turns a given link diagram D_1 into another link diagram D_2 , then D_1 and D_2 are called *equivalent*.

As was the case with link equivalence, link diagram equivalence allows the collection of all link diagrams to be partitioned into equivalence classes.

2B. Splittable link diagrams, non-self-intersections, and linking numbers. The ability to determine whether or not the components of a two-component link diagram can be separated from each other will be crucial for determining the winner of the linking-unlinking game. As such, we divide the family of link diagrams into splittable link diagrams and unsplittable link diagrams.

Definition 2.8. A link diagram is called *splittable* if it is equivalent to a *split link diagram* where one or more components of the diagram can be separated from one or more remaining components of the diagram by a circle. If a link diagram is not splittable (resp. not split), then it is called *unsplittable* (resp. *nonsplit*).

If a link diagram is splittable, we can imagine being able to split one or more components away from the rest of the link diagram. Figures 9 and 10 provide examples of splittable and unsplittable two-component link diagrams, respectively.

Now with an understanding of splittable and unsplittable link diagrams, we will introduce an invariant of two-component links called the linking number which will be used to help us detect when a link diagram is unsplittable. To define this quantity, we first need to discuss orientations of link pseudodiagrams.

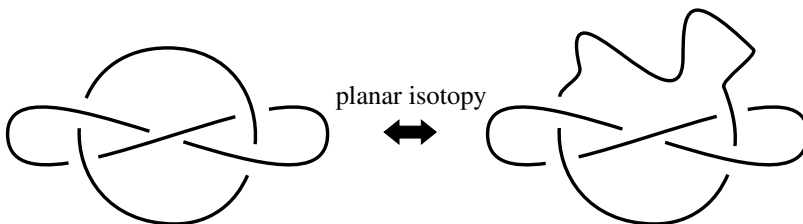


Figure 8. An example of a planar isotopy.

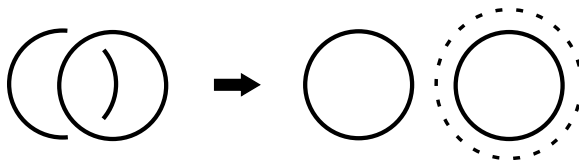


Figure 9. An example of a splittable two-component link diagram (where each link diagram component in this case is an unknotted circle). The dashed circle on the right separates the two unknotted components of the link diagram.

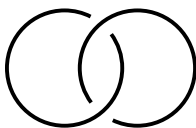


Figure 10. An example of an unsplittable two-component link diagram called the Hopf link diagram.

Definition 2.9. An *oriented* link pseudodiagram is a link pseudodiagram where one of two possible directions of travel has been chosen for each component of the link pseudodiagram.

For an example of an oriented two-component link diagram, see Figure 11. Given an oriented link pseudodiagram, we can associate signs to each resolved crossing.

Definition 2.10. To each resolved crossing of an oriented link pseudodiagram, we can associate a *crossing sign* as shown in Figure 12.

In Figure 11, each crossing is labeled with its sign. The final ingredient needed to define the linking number is the classification of the crossings of a link pseudodiagram into self-intersections and non-self-intersections.

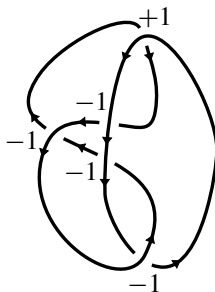


Figure 11. An oriented two-component link diagram labeled with the signs of each of its crossings. Arrows are used to denote the choice of direction of travel.

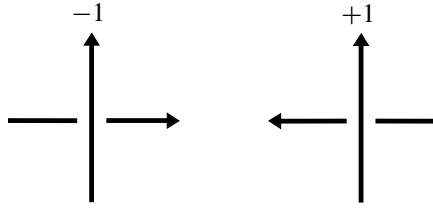


Figure 12. The left figure shows a -1 crossing and the right figure shows a $+1$ crossing.

Definition 2.11. If the two strands meeting at a crossing of a link pseudodiagram come from the same component of the link pseudodiagram, then such a crossing is called a *self-intersection* (SI). If a crossing is not a self-intersection, then we call it a *non-self-intersection* (NSI).

Note that all crossings of a knot pseudodiagram are necessarily SIs. Consequently, NSIs can only occur in link pseudodiagrams containing at least two components. Figure 13 provides examples of SIs and NSIs in a link diagram. The idea of classifying the crossings of a link pseudodiagram as SIs or NSIs will be used in the definition of the linking number below, as well as later on in this paper.

Definition 2.12. The *linking number* of an oriented two-component link diagram is defined to be half of the sum of the crossing signs, where the sum is taken over all of the non-self-intersections (NSIs) of the link diagram.

Looking back at Figure 11, we see that the two-component link diagram has a linking number of $\frac{1}{2}[3(-1) + 1(1)] = -1$. Observe that the leftmost crossing, which has crossing sign -1 , is not included in the linking number computation because this crossing is an SI. The linking number is an *invariant* of oriented two-component link diagrams, which means that two equivalent oriented two-component link diagrams have the same linking number. Note that any splittable two-component link diagram has linking number 0, as such a link diagram is equivalent to a link diagram with no NSIs. This tells us, by the contrapositive, that any oriented two-component link

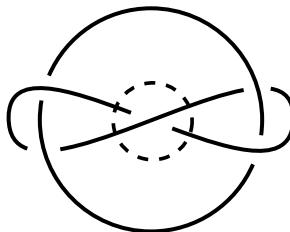


Figure 13. The dashed circle surrounds a crossing that is a self-intersection (SI). All other crossings are non-self-intersections (NSIs).

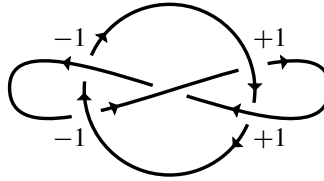


Figure 14. The Whitehead link is unsplittable with a linking number of 0.

diagram with nonzero linking number is unsplittable. An oriented two-component link diagram with linking number 0 is not necessarily splittable, however. An example of this is shown in Figure 14. To compute the linking number in this example, we ignore the center crossing because it is an SI. The two crossings on the left have a negative sign while the two crossings on the right have a positive sign. Summing these signs gives a linking number of $\frac{1}{2}[2(-1) + 2(1)] = 0$ but the link diagram is known to be unsplittable through other methods.

The following proposition, which will be useful later in this paper, determines the parity of the number of NSIs in a two-component link pseudodiagram.

Proposition 2.13. *Every two-component link pseudodiagram contains an even number of non-self-intersections (NSIs).*

Proof. The result is clearly true for a split two-component link pseudodiagram since it contains no NSIs. Now consider a nonsplit two-component link pseudodiagram (which necessarily contains NSIs). Choose a component of the link pseudodiagram and call it component 1. We can distinguish between the “inside” and “outside” of component 1 by giving it a canonical checkerboard coloring, that is, by coloring the regions of the shadow of component 1 either black or white so that regions sharing an edge have opposite colors and so that the unbounded region is colored white. We then view the black regions as the “inside” and the white regions as the “outside” of component 1. An example of a canonical checkerboard coloring is shown in Figure 15. It is a classical result that every link pseudodiagram has a checkerboard coloring. (For more details, see Section X.6 of [Bollobás 1998].)

Call the second component of the link pseudodiagram component 2. Assign component 2 an orientation and choose a starting point on component 2 that is outside of component 1. Since there are NSIs, at some point component 2 will cross from the outside of component 1 to the inside of component 1. Since component 2 enters the inside of component 1, it must also exit the inside of component 1 because component 2 is a closed curve that starts outside of component 1. This follows from the Jordan curve theorem, which states that every simple closed curve in the plane has an interior and an exterior. (For more details, see Chapter 3 of [Henle 1979].) Thus, the link pseudodiagram must contain at least two NSIs.

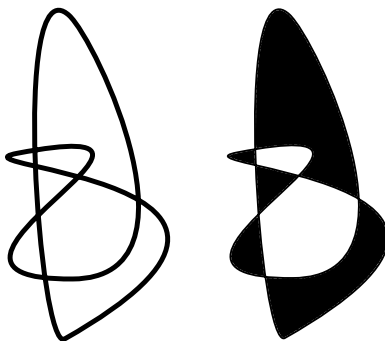


Figure 15. A knot shadow on the left with its canonical checkerboard coloring on the right. The black regions indicate the “inside” and the white regions indicate the “outside” of the knot shadow.

Following the orientation of component 2 from its starting point, the first NSI will bring us from the outside of component 1 to the inside of component 1. The second NSI will bring us back to the outside of component 1. By iterating this argument, we can see that we are outside component 1 after passing through an even number of NSIs and we are inside component 1 after passing through an odd number of NSIs.

Suppose the link pseudodiagram contained an odd number of NSIs. Then, by the argument from the preceding paragraph, we would end up inside component 1 after beginning at the starting point, following the orientation of component 2, and returning to the starting point. But then the endpoint of component 2 is inside component 1 and the starting point of component 2 is outside component 1. This is a contradiction since the starting point and the endpoint of component 2 are the same point. Therefore, there cannot be an odd number of NSIs in a two-component link pseudodiagram. $\square$

2C. Rational tangles and rational link diagrams. Rational links come from rational tangles that are formed by an iterative process of twisting two strands.

Definition 2.14. We define a *rational tangle*, denoted by $(a_1, a_2, \dots, a_n)$, where all of the a_i are integers, through the following construction. We begin with two parallel vertical strands, as shown in the top left of Figure 16. We then read $(a_1, a_2, \dots, a_n)$ from left to right, twisting strands as we go. In particular, we begin by twisting the bottom two endpoints $|a_1|$ times and then proceed to alternate between twisting the right two endpoints $|a_{2k}|$ times and twisting the bottom two endpoints $|a_{2k+1}|$ times. If $a_i > 0$ (resp. $a_i < 0$), then we apply $|a_i|$ twists in such a way that the overstrand has a positive (resp. negative) slope. We call the a_i the *syllables* of the *rational tangle word* $(a_1, a_2, \dots, a_n)$.

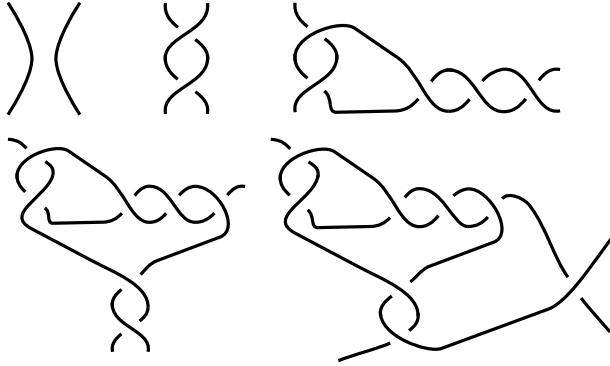


Figure 16. The rational tangles (0) (top left), (2) (top center), $(2, -3)$ (top right), $(2, -3, -2)$ (bottom left), and $(2, -3, -2, 1)$ (bottom right).

See Figure 16 for an example of the construction of a rational tangle. Since the linking-unlinking game will be played on two-component link pseudodiagrams, we need to generalize our notation for rational tangle words to allow for unresolved crossings.

Definition 2.15 [Johnson 2011]. By making a subset of the crossings of a rational tangle unresolved, we create a *rational pseudotangle*. We denote a rational pseudotangle by a *rational pseudotangle word* $(a_1(b_1), \dots, a_n(b_n))$, where the $a_i(b_i)$ are called the *syllables* of the word and where $|a_i|$ denotes the number of resolved crossings in the i -th syllable and $|b_i|$ denotes the number of unresolved crossings in the i -th syllable. If either $a_i = 0$ or $b_i = 0$ (but not both) for some i , then we omit the single occurrence of 0 or (0) in the rational pseudotangle word. If both $a_i = 0$ and $b_i = 0$ for some i , then we replace $a_i(b_i)$ by 0 in the rational pseudotangle word.

We now present a number of tangle equivalences that will be useful later in this paper.

Proposition 2.16 [Johnson 2011, Lemma 4.2]. *The following statements provide a set of tangle equivalences for rational tangles. Similar statements can also be made for rational pseudotangles:*

- (0) $(a_1, \dots, a_i, 0) = (a_1, \dots, a_i)$.
- (1) $(a_1, \dots, a_i, 0, a_{i+1}, \dots, a_n) = (a_1, \dots, a_i + a_{i+1}, \dots, a_n)$.
- (2) $(a_1, \dots, a_i, 0, 0, a_{i+1}, \dots, a_n) = (a_1, \dots, a_i, a_{i+1}, \dots, a_n)$.
- (3) $(0, a_1 + 1, a_2, \dots, a_n) = (0, a_1, a_2, \dots, a_n)$.
- (4) $(1, a_1, a_2, \dots, a_n) = (a_1 + 1, a_2, \dots, a_n)$.
- (5) $(-1, a_1, a_2, \dots, a_n) = (a_1 - 1, a_2, \dots, a_n)$.

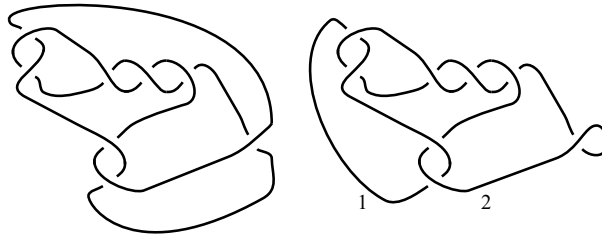


Figure 17. The numerator closure of the tangle $(2, -3, -2, 1)$ produces a knot diagram (left) and the denominator closure of the tangle $(2, -3, -2, 1)$ produces a two-component link diagram where the components have been numbered 1 and 2 (right).

Given that a rational pseudotangle has four endpoints, there are two ways to close a rational pseudotangle to form a rational link pseudodiagram.

Definition 2.17. A rational pseudotangle can be closed to form a *rational link pseudodiagram* either by connecting the top endpoints together and the bottom endpoints together (forming the *numerator closure*) or by connecting the left endpoints together and the right endpoints together (forming the *denominator closure*).

The left side of Figure 17 shows the numerator closure of the rational tangle $(2, -3, -2, 1)$ from Figure 16, which creates a knot diagram, and the right side of Figure 17 shows the denominator closure of this tangle, which creates a two-component link diagram. In general, either both closures of a rational pseudotangle will be knot pseudodiagrams or one closure will be a knot pseudodiagram and the other will be a two-component link pseudodiagram.

2D. Non-self-intersections for rational tangles. In Definition 2.11, we defined self-intersections (SIs) and non-self-intersections (NSIs) for link pseudodiagrams. We will now define SIs and NSIs for rational pseudotangles.

Definition 2.18. A crossing that occurs between a strand of a rational pseudotangle and itself is called a *self-intersection (SI)*. Otherwise, the crossing occurs between the two strands and is called a *non-self-intersection (NSI)*. Furthermore, a syllable of a rational pseudotangle word is called an *SI syllable* (resp. *NSI syllable*) if all of the crossings in the syllable are SIs (resp. NSIs).

If a rational pseudotangle has a closure that is a two-component link pseudodiagram, then the two strands of the pseudotangle become the two separate components of the link pseudodiagram and the SIs and NSIs of the pseudotangle become the SIs and NSIs of the link pseudodiagram, respectively. If a rational pseudotangle does not have a closure that is a two-component link pseudodiagram (if both closures result in a knot pseudodiagram), then both the SIs and the NSIs of the pseudotangle become SIs of the knot pseudodiagram since knot pseudodiagrams cannot contain NSIs.

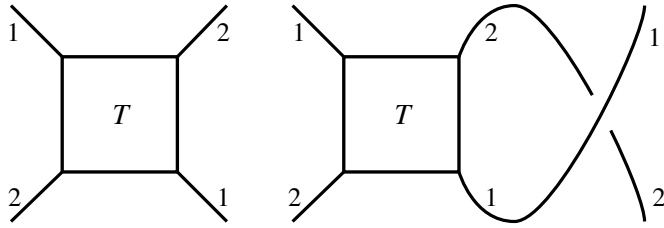


Figure 18. A rational pseudotangle T where strand 1 has endpoints in the northwest and southeast corners and strand 2 has endpoints in the northeast and southwest corners (left) and the result of adding a half-twist to this pseudotangle (right).

When looking for winning strategies for the linking-unlinking game played on rational two-component link shadows, we need to make sure that there actually exists a closure of the rational pseudotangle that is a two-component link pseudodiagram. The following result addresses this issue.

Proposition 2.19. *A rational pseudotangle has a closure that is a two-component link pseudodiagram if and only if the pseudotangle contains an even number of NSIs.*

Proof. ($\Rightarrow$) Assume we have a two-component link pseudodiagram that is a closure of a rational pseudotangle. By Proposition 2.13, the link pseudodiagram must contain an even number of NSIs. This implies that the rational pseudotangle must also contain an even number of NSIs because, otherwise, the rational pseudotangle would contain an odd number of NSIs and closure would create a two-component link pseudodiagram with an odd number of NSIs, contradicting Proposition 2.13.

($\Leftarrow$) Assume we have a rational pseudotangle that contains an even number of NSIs and suppose, for a contradiction, that both the numerator and denominator closures result in a knot pseudodiagram. The only way this can happen is for one strand of the pseudotangle to have endpoints in the northwest and southeast corners and the other strand to have endpoints in the northeast and southwest corners, as shown on the left side of Figure 18.

Add a half-twist to the end of the pseudotangle, as shown on the right side of Figure 18. This produces a rational pseudotangle with an odd total number of NSIs. We can see that the numerator closure of this new rational pseudotangle will result in a two-component link pseudodiagram. But this means we have a two-component link pseudodiagram with an odd total number of NSIs, which contradicts Proposition 2.13. $\square$

In the proposition that follows, we present information about SI syllables and NSI syllables for rational pseudotangle words.

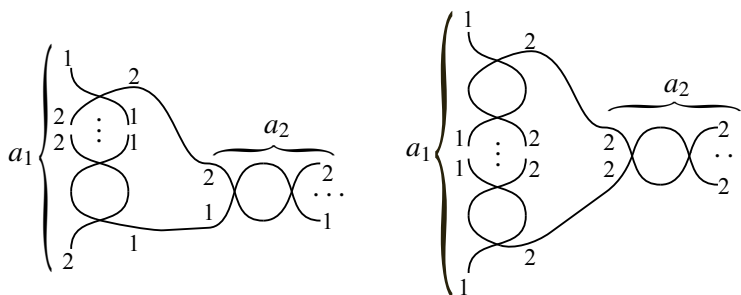


Figure 19. The case where a_1 is odd (left) and the case where a_1 is even (right). The two strands of the pseudotangle are numbered 1 and 2.

Proposition 2.20. *The following statements are true for rational tangle words $(a_1, \dots, a_n)$. Similar statements can also be made for rational pseudotangle words:*

- (1) *The first syllable a_1 is an NSI syllable.*
- (2) *The second syllable a_2 is an SI syllable if and only if a_1 is even.*
- (3) *If a_i is an SI syllable, then a_{i+1} is an NSI syllable.*
- (4) *Assume a_i is an SI syllable. Then a_{i+2} is an SI syllable if and only if a_{i+1} is even.*
- (5) *If both a_i and a_{i+1} are NSI syllables and a_{i+1} is odd, then a_{i+2} is an SI syllable.*
- (6) *If both a_i and a_{i+1} are NSI syllables and a_{i+1} is even, then a_{i+2} is an NSI syllable.*

Proof. Condition (1) is true by Definition 2.18 because the crossings of the first syllable a_1 are necessarily between the two strands of the rational tangle. The proof of the forward direction of condition (2) proceeds by contraposition and follows from the left side of Figure 19. The proof of the reverse direction of condition (2) follows from the right side of Figure 19.

For the remainder of this proof, let L denote the rational tangle $(a_1, a_2, \dots, a_{i-1})$ that precedes the syllable a_i . Note that when we construct a rational tangle, the northwest endpoint of L remains fixed. Label the strand of L incident to this endpoint strand 1 and label the other strand of L strand 2. Furthermore, we can assume that a_i is a horizontal twist without loss of generality, as we can create the case where a_i is a vertical twist by reflecting the diagram over the line $y = -x$.

To prove condition (3), assume a_i is an SI syllable. Then, as shown in Figure 20, the next syllable a_{i+1} will be an NSI syllable. For the remaining three conditions, we will consider cases. Figures 21, 22, and 23 show these six total cases, grouped into three pairs based on the location of the second endpoint of strand 1 of L and

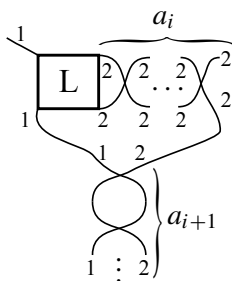


Figure 20. An SI syllable a_i followed by an NSI syllable a_{i+1} .

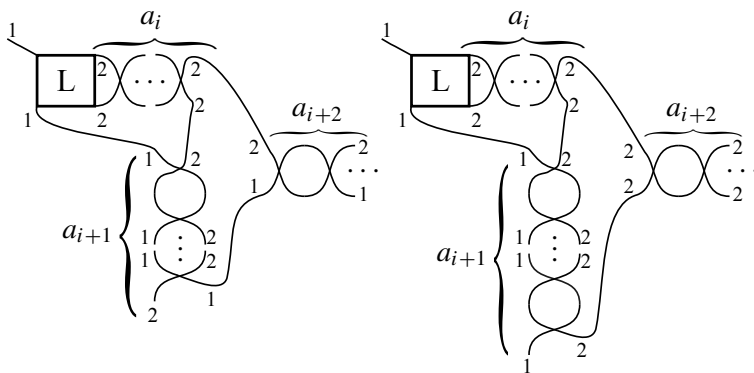


Figure 21. The two cases where the southwest endpoint of L comes from strand 1.

based on the parity of a_{i+1} . To prove condition (4), assume a_i is an SI syllable. The proof of the forward direction of condition (4) proceeds by contraposition and follows from the left side of Figure 21. The proof of the reverse direction of condition (4) follows from the right side of Figure 21. To prove conditions (5) and (6) (combined), assume both a_i and a_{i+1} are NSI syllables. The proof of these conditions follows from Figures 22 and 23.

□

We now present notation used to highlight the presence of isolated SI syllables in rational tangle words.

Notation 2.21. Given a syllable a of a rational tangle word, we use a^* to indicate that the syllable a is an SI syllable.

Given Proposition 2.20, we now present a result that leads to a method to decompose any rational pseudotangle word into SI syllables and strings of NSI syllables.

Proposition 2.22. *Every rational tangle word can be decomposed into strings of NSI syllables that alternate with isolated SI syllables. Furthermore:*

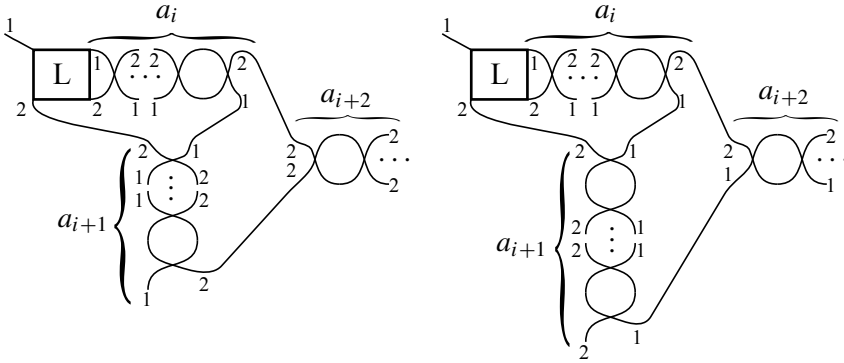


Figure 22. The two cases where the northeast endpoint of L comes from strand 1 and where a_i and a_{i+1} are NSI syllables. Note that a_i must be odd for a_{i+1} to be an NSI syllable.

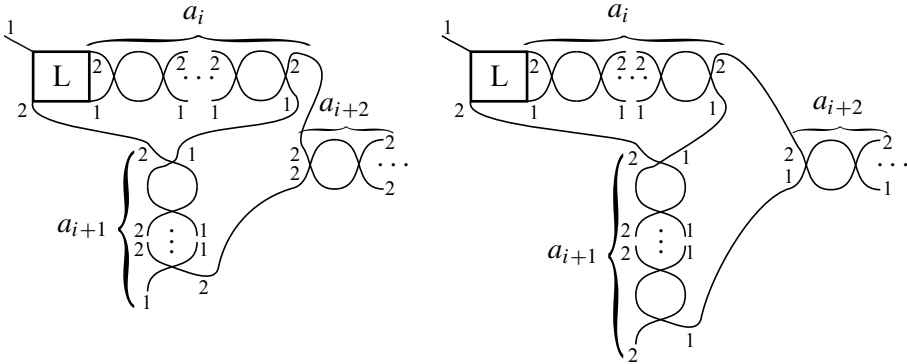


Figure 23. The two cases where the southeast endpoint of L comes from strand 1 and where a_i and a_{i+1} are NSI syllables. Note that a_i must be even for a_{i+1} to be an NSI syllable.

- (1) *All but the last string of NSI syllables consists of either*
 - *a single even syllable,*
 - *two consecutive odd syllables, or*
 - *an odd syllable followed by an arbitrary nonempty string of even syllables followed by a final odd syllable.*
- (2) *If the rational tangle contains an even number of NSIs, then the last string of NSI syllables consists of the same possibilities as condition (1).*
- (3) *If the rational tangle contains an odd number of NSIs, then the last string of NSI syllables consists of either*
 - *a single odd syllable, or*
 - *an odd syllable followed by an arbitrary nonempty string of even syllables.*

A similar statement can also be made for rational pseudotangle words.

Proof. By Proposition 2.20(3), no two SI syllables can be adjacent. Thus, the strings of NSI syllables alternate with isolated SI syllables.

We will now prove condition (1). Suppose we have a nonfinal string of NSI syllables. We want to show that this string consists of a single even syllable or an odd syllable followed by an arbitrary (possibly empty) string of even syllables followed by a final odd syllable. In Cases 1 and 2 below, we consider the first nonfinal string of NSI syllables, whose first syllable is a_1 . By Proposition 2.20(1), we know that a_1 is an NSI syllable.

Case 1: Suppose a_1 is even. Then Proposition 2.20(2) implies that a_2 is an SI syllable, so the first nonfinal string of NSI syllables consists of the single even syllable a_1 .

Case 2: Suppose a_1 is odd. Then Proposition 2.20(2) implies that a_2 is an NSI syllable. If a_2 is odd, then Proposition 2.20(5) implies that a_3 is an SI syllable, which gives a string of NSI syllables composed of two odd syllables. If a_2 is even, then Proposition 2.20(6) implies that a_3 is an NSI syllable. This string of even NSI syllables continues by repeatedly applying Proposition 2.20(6). This string of even NSI syllables must eventually terminate, however, since this is a nonfinal string of NSI syllables. Therefore, we eventually find an odd syllable in this string of NSI syllables; call this syllable a_k . This syllable a_k is the last syllable in this string of NSI syllables because the next syllable a_{k+1} is an SI syllable by Proposition 2.20(5). Thus, the first nonfinal string of NSI syllables consists of an odd syllable a_1 followed by an arbitrary (possibly empty) string $a_2, \dots, a_{k-1}$ of even syllables followed by a final odd syllable a_k .

In Cases 3 and 4 below, we consider a nonfirst nonfinal string of NSI syllables. Let a_i , where $i > 1$, denote the first syllable in this string of NSI syllables.

Case 3: Suppose a_i is even. Since a_i is the first NSI syllable in a nonfirst string of NSI syllables, we know that a_{i-1} is an SI syllable. By Proposition 2.20(4), a_{i+1} is an SI syllable, so the nonfirst nonfinal string of NSI syllables consists of the single even syllable a_i .

Case 4: Suppose a_i is odd. Since a_i is the first NSI syllable in a nonfirst string of NSI syllables, we know that a_{i-1} is an SI syllable. Then Proposition 2.20(4) implies that a_{i+1} is an NSI syllable. The remainder of this case is similar to Case 2, except that a_{i+1} is now playing the role of a_2 .

This completes the proof of condition (1). To prove condition (2) and condition (3), we will consider the final string of NSI syllables. Let a_i be the first syllable in this final string of NSI syllables. We will consider two cases based on the parity of the NSIs.

To prove condition (2), suppose there are an even number of NSIs in the tangle word. Condition (1) of this proposition implies that each nonfinal string of NSI syllables contains an even number of NSIs. Thus, the final string of NSI syllables must contain an even number of NSIs for the tangle word to contain an even total number of NSIs.

Case A: Suppose a_i is even. If the final string of NSI syllables consists of the syllable a_i alone, then we have the desired result. We may now assume that the final string of NSI syllables contains a second syllable a_{i+1} . If the final string of NSI syllables is the only string of NSI syllables in the tangle word, then $a_i = a_1$, $a_{i+1} = a_2$, and the argument from Case 1 gives the desired result. Now suppose there are at least two strings of NSI syllables in the tangle word. Then the argument from Case 3 gives the desired result.

Case B: Suppose a_i is odd. If the final string of NSI syllables consists of the syllable a_i alone, then we have a contradiction of the fact that the final string of NSI syllables contains an even number of NSIs. We may now assume that the final string of NSI syllables contains a second syllable a_{i+1} .

Subcase 1: Suppose the final string of NSI syllables is the only string of NSI syllables in the tangle word. Then $a_i = a_1$ and $a_{i+1} = a_2$.

Suppose a_2 is odd. If the final string of NSI syllables contains only a_1 and a_2 , then we have the desired result. If the final string of NSI syllables contains a third syllable a_3 , then the argument from the first three sentences of Case 2 gives the desired result.

Suppose a_2 is even. If the final string of NSI syllables contains only a_1 and a_2 , then we have a contradiction of the fact that the final string of NSI syllables contains an even number of NSIs. If the final string of NSI syllables contains a third syllable a_3 , then the argument from Case 2 implies that a_1 is followed by a nonempty string of even syllables. Eventually, an odd syllable must occur because the total number of NSIs in the final string of NSI syllables must be even. Then either this odd syllable is the last syllable of the tangle word or the next syllable is an SI syllable by Proposition 2.20(5). In either case, we have the desired result.

Subcase 2: Now suppose there are at least two strings of NSI syllables in the tangle word.

Suppose a_{i+1} is odd. If the final string of NSI syllables contains only a_i and a_{i+1} , then we have the desired result. If the final string of NSI syllables contains a third syllable a_{i+2} , then the argument from Case 4 gives the desired result.

Suppose a_{i+1} is even. If the final string of NSI syllables contains only a_i and a_{i+1} , then we have a contradiction of the fact that the final string of NSI syllables contains an even number of NSIs. If the final string of NSI syllables contains a third syllable a_{i+2} , then the argument from Case 4 implies that a_i is followed by a

nonempty string of even syllables. Eventually, an odd syllable must occur because the total number of NSIs in the final string of NSI syllables must be even. Then either this odd syllable is the last syllable of the tangle word or the next syllable is an SI syllable by Proposition 2.20(5). In either case, we have the desired result.

To prove condition (3), suppose there are an odd number of NSIs in the tangle word. Condition (1) of this proposition implies that each nonfinal string of NSI syllables contains an even number of NSIs. Thus, the final string of NSI syllables must contain an odd number of NSIs for the tangle word to contain an odd total number of NSIs.

Case A: Suppose a_i is even. If the final string of NSI syllables consists of the syllable a_i alone, then we have a contradiction of the fact that the final string of NSI syllables contains an odd number of NSIs. We may now assume that the final string of NSI syllables contains a second syllable a_{i+1} . If the final string of NSI syllables is the only string of NSI syllables in the tangle word, then $a_i = a_1$, $a_{i+1} = a_2$, and the argument from Case 1 gives a contradiction of the fact that the final string of NSI syllables contains an odd number of NSIs. Now suppose there are at least two strings of NSI syllables in the tangle word. Then the argument from Case 3 gives a contradiction of the fact that the final string of NSI syllables contains an odd number of NSIs.

Case B: Suppose a_i is odd. If the final string of NSI syllables consists of the syllable a_i alone, then we have the desired result. We may now assume that the final string of NSI syllables contains a second syllable a_{i+1} .

Subcase 1: Suppose the final string of NSI syllables is the only string of NSI syllables in the tangle word. Then $a_i = a_1$ and $a_{i+1} = a_2$.

Suppose a_2 is odd. If the final string of NSI syllables contains only a_1 and a_2 , then we have a contradiction of the fact that the final string of NSI syllables contains an odd number of NSIs. If the final string of NSI syllables contains a third syllable a_3 , then the argument from the first three sentences of Case 2 gives a contradiction of the fact that the final string of NSI syllables contains an odd number of NSIs.

Suppose a_2 is even. If the final string of NSI syllables contains only a_1 and a_2 , then we have the desired result. If the final string of NSI syllables contains a third syllable a_3 , then the argument from Case 2 implies that a_1 is followed by a nonempty string of even syllables which must terminate because the tangle word is finite. This gives the desired result.

Subcase 2: Now suppose there are at least two strings of NSI syllables in the tangle word.

Suppose a_{i+1} is odd. If the final string of NSI syllables contains only a_i and a_{i+1} , then we have a contradiction of the fact that the final string of NSI syllables contains an odd number of NSIs. If the final string of NSI syllables contains a third

syllable a_{i+2} , then the argument from Case 4 gives a contradiction of the fact that the final string of NSI syllables contains an odd number of NSIs.

Suppose a_{i+1} is even. If the final string of NSI syllables contains only a_i and a_{i+1} , then we have the desired result. If the final string of NSI syllables contains a third syllable a_{i+2} , then the argument from Case 4 implies that a_i is followed by a nonempty string of even syllables which must terminate because the tangle word is finite. This gives the desired result. $\square$

We now present an example of applying Proposition 2.22 to decompose a rational tangle word into SI syllables and NSI syllables.

Example 2.23. Consider the rational tangle word

$$(1, 4, 2, 1, 3, 5, 3, 2, 1, 2, 0, 5, 2, 6, 4).$$

Reading from left to right, we see that our first string of NSI syllables is 1, 4, 2, 1, as this is an odd syllable followed by a nonempty string of even syllables followed by a final odd syllable. The next syllable, 3, is an SI syllable. We now update our tangle word to get

$$(1, 4, 2, 1, 3^*, 5, 3, 2, 1, 2, 0, 5, 2, 6, 4).$$

Proceeding to the right, we see that 5, 3 is the next string of NSI syllables, as this is a string of two consecutive odd syllables. The next syllable, 2, is an SI syllable, so we update our tangle word to get

$$(1, 4, 2, 1, 3^*, 5, 3, 2^*, 1, 2, 0, 5, 2, 6, 4).$$

Continuing this process, we see that 1, 2, 0, 5 is the next string of NSI syllables and the following syllable, 2, is an SI syllable. Updating our tangle word, we get

$$(1, 4, 2, 1, 3^*, 5, 3, 2^*, 1, 2, 0, 5, 2^*, 6, 4).$$

The final string of NSI syllables consists of the single even syllable 6, so our rational tangle word ends with the syllable 4, which is an SI syllable. Thus, our completely decomposed tangle word is given by

$$(1, 4, 2, 1, 3^*, 5, 3, 2^*, 1, 2, 0, 5, 2^*, 6, 4^*).$$

3. The linking-unlinking game

In this section, we will define the linking-unlinking game and provide winning strategies for the linking-unlinking game played on all shadows of two-component rational tangle closures and played on a large family of general two-component link shadows.

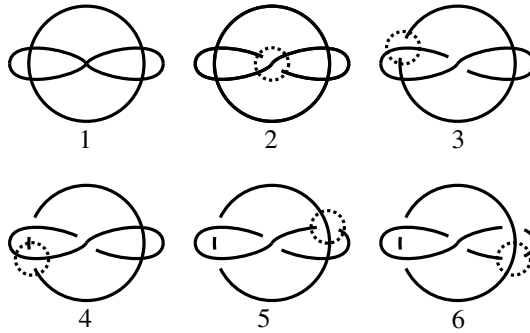


Figure 24. The linking-unlinking game played on a shadow of the Whitehead link. Crossings being played on are surrounded by a dotted circle.

3A. Defining the linking-unlinking game. In what follows, we introduce the linking-unlinking game and present two key player strategies that will be used often in the remainder of this paper.

Definition 3.1. The *linking-unlinking game* is a two-player game played on the shadow of a two-component link. The game is played with each player taking turns resolving crossings of the link shadow. The game ends when all of the crossings are resolved and a two-component link diagram is formed. One player, the *linker*, wins if the resulting link diagram is unsplittable. The other player, the *unlinker*, wins if the resulting link diagram is splittable.

An example of the linking-unlinking game is provided below.

Example 3.2. Figure 24 shows an example of the linking-unlinking game being played on a shadow of the Whitehead link. Here, the unlinker goes first. The second frame shows that the unlinker decides to play on the central crossing. Next, the linker plays on the upper left crossing, as shown in the third frame. In the fourth frame, the unlinker responds on the lower left crossing, resolving the crossing with the intention of being able to reduce the two left crossings with an R2 move. Next, the linker plays on the top right crossing, as shown in the fifth frame. Finally, the sixth frame shows that the unlinker responds on the lower right crossing, resolving the crossing with the intention of being able to reduce the two right crossings with an R2 move. The link diagram at the end of the game is equivalent to the splittable trivial two-component link diagram (also called the two-component unlink diagram), so the unlinker wins. Figure 25 shows this equivalence.

Given the definition of the linking-unlinking game as a two-player combinatorial game, our main goal will now be to determine who wins the game and who loses the game for various two-component link shadows.

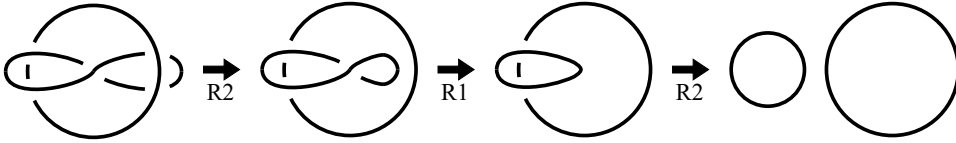


Figure 25. A sequence of Reidemeister moves used to show that the two-component link diagram above on the left is splittable.

Definition 3.3. In a two-player game, a player has a *winning strategy* if there is always a sequence of moves that allows them to win, regardless of what their opponent does.

Given the structure of rational pseudotangles, we will often group together crossings in a syllable of the pseudotangle word when constructing winning strategies for the linking-unlinking game.

Definition 3.4. A *subtwist region* in a link pseudodiagram is a section of the associated link shadow consisting of a chain of bigons. A *twist region* in a link pseudodiagram is a section of the associated link shadow consisting of a maximal chain of bigons, maximal in the sense that the chain cannot be extended by adding more bigons. Note that a twist region consisting of a single crossing with no incident bigons is possible.

A chain of bigons can be thought of as being formed by twisting two parallel strands. Figure 26 provides a schematic depiction of a twist region. Given a subtwist region of a link pseudodiagram, we now define an operation called *flying* that will be useful in defining two important player strategies for the linking-unlinking game.

Definition 3.5. A *flype* is a move performed on a tangle of a link pseudodiagram that reflects the tangle over an axis.

See Figure 27 for an example of a flype. For our purposes, we will only consider flypes applied to subtwist regions of a tangle pseudodiagram. In what follows, we present two key player strategies for the linking-unlinking game that will be applied

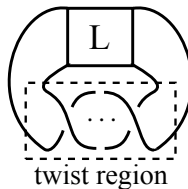


Figure 26. A schematic diagram where the twist region is surrounded by a dashed rectangle. The box labeled L represents the remainder of the link pseudodiagram.

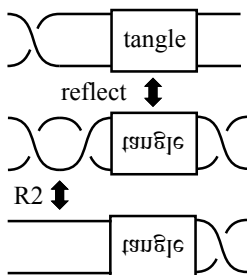


Figure 27. A schematic diagram of a flype applied to a tangle in a link pseudodiagram.

throughout the remainder of this paper. Note that these are response strategies for the next player regardless of the role of the current player.

Strategy 3.6 (R2-strategy for the next player). Assign the given link pseudodiagram an orientation. When a player plays by resolving a crossing in a twist region (consequently assigning this crossing a crossing sign), the next player responds by resolving a crossing in the same twist region so that this crossing has crossing sign opposite to the previous resolved crossing. This allows both crossings to be removed by applying a flype and an R2 move, as shown in Figure 28.

The R2-strategy is useful for both players because it allows for pairs of crossings to be removed at the end of the game.

Strategy 3.7 (anti-R2-strategy for the next player). Assign the given link pseudodiagram an orientation. Here, when a player plays by resolving a crossing in a twist region, the next player responds by resolving a crossing in the same twist region

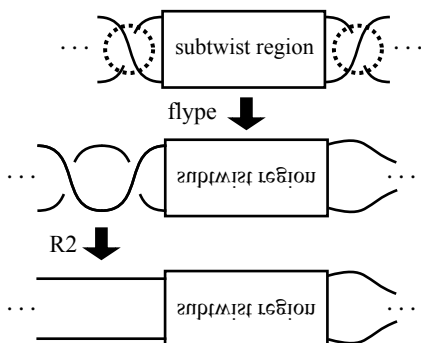


Figure 28. An example of the R2-strategy. A player resolves a crossing and the next player responds on another crossing in the twist region so that a flype and an R2 move will remove the pair of crossings.

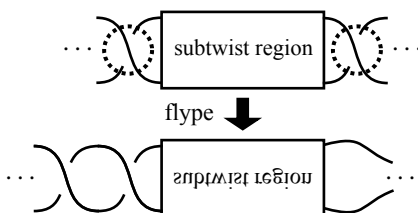


Figure 29. An example of the anti-R2-strategy. A player resolves a crossing and the next player responds on another crossing in the twist region so that a flype will create a clasp between the two strands of the twist region.

so that this crossing has the same crossing sign as the previous resolved crossing. In this situation, the two crossings cannot be removed by applying a flype to the subtwist region and one of the two incident crossings, as shown in Figure 29. Note that one or both of these crossings may be able to be removed, however, by applying Reidemeister moves to the remainder of the link diagram at the end of the game.

The anti-R2-strategy is particularly useful for the linker because it creates a clasp between two strands. As an example of why creating clasps might be useful for the linker, consider the unsplittable Hopf link diagram in Figure 10.

3B. The linking-unlinking game for rational two-component link shadows. In this section, we provide winning strategies for the linking-unlinking game played on shadows of two-component rational tangle closures. We begin by providing a lemma that will be used in the proofs of a number of theorems in the remainder of this paper.

Lemma 3.8. *The following statements are true:*

- (1) *The rational tangle (0) has a splittable two-component link closure.*
- (2) *The rational tangle (± 2) has an unsplittable two-component link closure.*

Proof. To prove statement (1), observe from the top left of Figure 16 that the denominator closure of the tangle (0) gives the trivial two-component link diagram. This link diagram is clearly splittable.

To prove statement (2), observe that the denominator closure of the tangle (± 2) gives either the Hopf link diagram from Figure 10 or its reflection. In either case, after orienting this link diagram, it can be seen that the linking number of the resulting oriented two-component link diagram has absolute value 1. Since this link diagram has nonzero linking number, it is unsplittable. $\square$

We now begin our study of the linking-unlinking game by focusing on certain families of rational two-component link shadows. The following result presents winning strategies for the unlinker in a very specific pathological case and for the second player in other cases.

Theorem 3.9. *Suppose we have a shadow of a rational two-component link coming from a closure of the rational tangle $(a_1, \dots, a_n)$:*

- (1) *If $a_{2k+1} = 0$ for all k , then the unlinker wins.*
- (2) *If $a_{2k+1} \neq 0$ for at least one k and all of the a_i are even, then the second player has a winning strategy.*

Proof. We begin by proving statement (1). Assume we have a shadow of a rational two-component link coming from a closure of the rational tangle $(a_1, \dots, a_n)$ where $a_{2k+1} = 0$ for all k . If $n = 1$, then the tangle word is (0) and the unlinker wins by Lemma 3.8. Now suppose that $n \geq 2$. Then the tangle word can be written in the form $(0, a_2, \dots, 0, a_{2m})$ or $(0, a_2, \dots, 0, a_{2m}, 0)$. By repeatedly applying Proposition 2.16(3), we can reduce each of the tangle subwords $(0, a_{2k})$ to $(0, 0)$. By repeatedly applying Proposition 2.16(2), we can remove all but the last copy of $(0, 0)$ from the tangle word. At the end of this process, what remains is either $(0, 0)$ or $(0, 0, 0)$, both of which are equivalent to (0) by applying Proposition 2.16(0) either once or twice, respectively. By Lemma 3.8, the unlinker wins.

We will now prove statement (2). Assume we have a shadow of a rational two-component link coming from a closure of the rational tangle $(a_1, \dots, a_n)$ where $a_{2k+1} \neq 0$ for at least one k and where all of the a_i are even. Fix a value j such that $a_{2j+1} \neq 0$. We consider two cases, depending on the role of the second player.

Case 1: Suppose the unlinker plays second. Whenever the linker plays on a syllable, the unlinker should respond on the same syllable by using the R2-strategy. Note that this response strategy is always possible since all of the a_i are even. Thus, each time the unlinker responds, two crossings can be removed at the end of the game. After applying flypes and R2 moves at the end of the game, the resulting tangle word will be of the form $(0, \dots, 0)$. By repeatedly applying Proposition 2.16(0), this tangle word is equivalent to (0) and the unlinker wins by Lemma 3.8.

Case 2: Suppose the linker plays second. Whenever the unlinker plays on any a_i -syllable where $i \neq 2j + 1$, the linker should respond on this syllable by using the R2-strategy. Again, this strategy can be applied since each a_i is even. Thus, each time the linker responds, two crossings can be removed at the end of the game. When the unlinker plays on $a_{2j+1} \neq 0$, the linker should respond by using the R2-strategy until only two unresolved crossings remain in this subtwist region. On the last two unresolved crossings in this subtwist region, the linker should respond by using the anti-R2-strategy.

After applying flypes and R2 moves at the end of the game, the resulting tangle word will be of the form $(0, \dots, 0, 2, 0, \dots, 0)$ or $(0, \dots, 0, -2, 0, \dots, 0)$, where there are an even number of zeros before the 2 or -2 . By repeatedly applying Proposition 2.16(2), we can reduce the tangle word to either $(2, 0, \dots, 0)$ or

$(-2, 0, \dots, 0)$. By repeatedly applying Proposition 2.16(0), these tangle words are equivalent to either (2) or (-2) , respectively, and the linker wins by Lemma 3.8. $\square$

We now present winning strategies for rational two-component link shadows whose tangle word consists of two odd syllables.

Theorem 3.10. *Playing on the shadow of a rational two-component link coming from a closure of the rational tangle (a_1, a_2) , if a_1 and a_2 are both odd, then the second player has a winning strategy.*

Proof. Assume we have a shadow of a rational two-component link coming from a closure of the rational tangle (a_1, a_2) where a_1 and a_2 are both odd. We will consider two cases, depending on the role of the second player.

Case 1: Suppose the linker plays second. When the unlinker plays, the linker should respond on the same syllable using the R2-strategy so long as this is possible. Since the unlinker is playing first and since both of the syllables are odd, the unlinker will necessarily have to resolve the last crossing in either a_1 or a_2 . Once the unlinker fully resolves an a_i -syllable, the pseudotangle word can be reduced to one of $(1, (a))$, $(-1, (a))$, $((a), 1)$, or $((a), -1)$, where a is odd. The linker should then play on a by resolving a crossing to have the same overcrossing slope as the previous crossing resolved by the unlinker. Thus, the linker will turn $(1, (a))$ into $(1, 1(|a| - 1))$, $(-1, (a))$ into $(-1, -1(|a| - 1))$, $((a), 1)$ into $(1(|a| - 1), 1)$, or $((a), -1)$ into $(-1(|a| - 1), -1)$.

Since $|a| - 1$ is even, the linker should resume using the R2-strategy to respond to the unlinker until the end of the game. This results in a rational tangle word that can be reduced to either $(1, 1)$ or $(-1, -1)$, which is equivalent to either (2) or (-2) by applying statement (4) or (5) of Proposition 2.16, respectively. By Lemma 3.8, the linker wins.

Case 2: Suppose the unlinker plays second. The unlinker's strategy will be similar to the linker's strategy from Case 1. First, the unlinker should use the R2-strategy until the linker fully resolves an a_i -syllable. As in the previous case, the pseudotangle word can be reduced to the form $(1, (a))$, $(-1, (a))$, $((a), 1)$, or $((a), -1)$, where a is odd. The unlinker should then play on a by resolving a crossing to have overcrossing slope opposite to the previous crossing resolved by the linker. Thus, the unlinker will turn $(1, (a))$ into $(1, -1(|a| - 1))$, $(-1, (a))$ into $(-1, 1(|a| - 1))$, $((a), 1)$ into $(-1(|a| - 1), 1)$, or $((a), -1)$ into $(1(|a| - 1), -1)$.

Since $|a| - 1$ is even, the unlinker should resume using the R2-strategy to respond to the linker until the end of the game. This results in a rational tangle word that can be reduced to either $(1, -1)$ or $(-1, 1)$, both of which are equivalent to (0) by applying statement (4) or (5) of Proposition 2.16, respectively. By Lemma 3.8, the unlinker wins. $\square$

The following theorem builds on the results of the previous two theorems.

Theorem 3.11. *Playing on the shadow of a rational two-component link coming from a closure of the rational tangle $(a_1, \dots, a_n)$ where $n \geq 3$, if a_1 and a_n are odd and all other a_i are even, then the second player has a winning strategy.*

Proof. Assume we have a shadow of a rational two-component link coming from a closure of the rational tangle $(a_1, \dots, a_n)$ where $n \geq 3$, where a_1 and a_n are odd, and where all other a_i are even. When the first player plays on a_1 or a_n , the second player should respond by using the same strategy as used in the proof of Theorem 3.10, as though the game were played on (a_1, a_n) , where the linker wants to reduce (a_1, a_n) to either $(1, 1)$ or $(-1, -1)$ and the unlinker wants to reduce (a_1, a_n) to either $(1, -1)$ or $(-1, 1)$. When the first player plays on one of the even a_i -syllables, the second player should respond on the same syllable by using the R2-strategy. At the end of the game, there are two cases for the resulting tangle word, depending on the role of the second player.

Case 1: Suppose the linker plays second. Then the resulting tangle word can be reduced to either $(1, 0, \dots, 0, 1)$ or $(-1, 0, \dots, 0, -1)$. If there is an even number of zeros, then the tangle word can be reduced to either $(1, 1)$ or $(-1, -1)$ by repeatedly applying Proposition 2.16(2) and reduced to either (2) or (-2) by applying statement (4) or (5) of Proposition 2.16, respectively. If there is an odd number of zeros, then the tangle word can be reduced to either $(1, 0, 1)$ or $(-1, 0, -1)$ by repeatedly applying Proposition 2.16(2) and reduced to either (2) or (-2) by applying Proposition 2.16(1). By Lemma 3.8, the linker wins.

Case 2: Suppose the unlinker plays second. Then the resulting tangle word can be reduced to either $(1, 0, \dots, 0, -1)$ or $(-1, 0, \dots, 0, 1)$. If there is an even number of zeros, then the tangle word can be reduced to either $(1, -1)$ or $(-1, 1)$ by repeatedly applying Proposition 2.16(2) and reduced to (0) by applying statement (4) or (5) of Proposition 2.16, respectively. If there is an odd number of zeros, then the tangle word can be reduced to either $(1, 0, -1)$ or $(-1, 0, 1)$ by repeatedly applying Proposition 2.16(2) and reduced to (0) by applying Proposition 2.16(1). By Lemma 3.8, the unlinker wins. $\square$

We will now explore how the parity of the number of SIs in the rational two-component link shadow affects which player has a winning strategy. The following key theorem, when paired with Theorem 3.9(1), provides winning strategies for all shadows of two-component rational tangle closures. Its proof combines a number of results from earlier in this paper.

Theorem 3.12. *Suppose we have a shadow of a rational two-component link coming from a closure of the rational tangle $(a_1, \dots, a_n)$. Furthermore, assume $a_{2k+1} \neq 0$ for some k :*

- (1) *If the tangle word contains an even number of SIs, then the second player has a winning strategy.*
- (2) *If the tangle word contains an odd number of SIs, then the first player has a winning strategy.*

Proof. Assume we have a shadow of a rational two-component link coming from a closure of the rational tangle $(a_1, \dots, a_n)$ where $a_{2k+1} \neq 0$ for some k .

We begin by proving statement (1). Assume $(a_1, \dots, a_n)$ contains an even number of SIs. Using Proposition 2.22, we can decompose $(a_1, \dots, a_n)$ into alternating strings of NSI syllables and isolated SI syllables. Whenever the first player plays on an SI, the second player should respond by playing arbitrarily on any other SI. Note that this response strategy is always possible since the number of SIs is even.

By Proposition 2.19, since we have a two-component link shadow coming from a closure of a rational tangle, the tangle must contain an even number of NSIs. Therefore, by Proposition 2.22, we have that a nonfinal or final string of NSI syllables can be either (a) a single even syllable, (b) two consecutive odd syllables, or (c) an odd syllable followed by an arbitrary nonempty string of even syllables followed by a final odd syllable. Notice that the proofs of Theorems 3.9(2), 3.10, and 3.11 provide the second player with a strategy for playing on the respective type (a), type (b), and type (c) strings of NSI syllables listed above.

When the first player plays on an unresolved crossing from a nonfinal string of NSI syllables, the second player should respond on an unresolved crossing in the same nonfinal string of NSI syllables by using the strategy for when the unlinker plays second provided by the proof of the theorem corresponding to the type of string of NSI syllables. This strategy ensures that each nonfinal string of NSI syllables can be reduced to the tangle subword (0).

When the first player plays on an unresolved crossing from the final string of NSI syllables, the second player should respond on an unresolved crossing in the same final string of NSI syllables by using the strategy that corresponds to their role as linker or unlinker from the proof of the theorem corresponding to the type of string of NSI syllables.

Note that the second player will always be able to respond on the same nonfinal or final string of NSI syllables because each such string contains an even total number of crossings. We will now consider two cases, depending on the role of the second player.

Case 1: Suppose the unlinker plays second. Let $(a_1, \dots, a_k)$ denote the first string of NSI syllables. By the proofs of Theorems 3.9(2), 3.10, and 3.11, we know that the unlinker's strategy results in this string of syllables being able to be reduced to the tangle subword (0).

Subcase 1: Suppose the entire tangle word contains a single string of NSI syllables. If the entire tangle word consists of a single string of NSI syllables without an SI syllable at the end of the tangle word, then the tangle word can be reduced to (0) and the unlinker wins by Lemma 3.8. If the entire tangle word consists of a single string of NSI syllables followed by an SI syllable, call it a^* , at the end of the tangle word, then the tangle word can be reduced to $(0, a^*)$, which can be reduced to $(0, 0)$ by repeatedly applying Proposition 2.16(3). By applying Proposition 2.16(0), $(0, 0)$ can be reduced to (0) and the unlinker wins by Lemma 3.8.

Subcase 2: Suppose the tangle word contains multiple strings of NSI syllables. Then we can apply the strategy from all but the last sentence of Subcase 1 to each tangle subword $(a_{k+1}, a_{k+2}, \dots, a_m, b_m^*)$, where $(a_{k+1}, a_{k+2}, \dots, a_m)$ is a nonfinal string of NSI syllables and where b_m^* is an SI syllable, so that each such subword can be reduced to $(0, 0)$. These subwords can then be removed from the tangle word by applying Proposition 2.16(2). If the final string of NSI syllables is not followed by a final SI syllable, then the tangle word can be reduced to (0) and the unlinker wins by Lemma 3.8. If the final string of NSI syllables is followed by a final SI syllable, then the argument from Subcase 1 can be applied in its entirety to conclude that the unlinker wins by Lemma 3.8.

Case 2: Suppose the linker plays second. Then, as indicated in the paragraphs preceding Case 1, the only difference in strategy will occur in the final string of NSI syllables. Specifically, the linker (and the unlinker) will respond so that all of the nonfinal strings of NSI syllables can be reduced to the tangle subword (0) and the linker will respond so that the final string of NSI syllables can be reduced to either the tangle subword (2) or the tangle subword (-2) .

Subcase 1: Suppose the entire tangle word contains a single string of NSI syllables. If the entire tangle word consists of a single string of NSI syllables without an SI syllable at the end of the tangle word, then the tangle word can be reduced to (2) or (-2) and the linker wins by Lemma 3.8. If the entire tangle word consists of a single string of NSI syllables followed by an SI syllable, call it a^* , at the end of the tangle word, then the tangle word can be reduced to $(2, a^*)$ or $(-2, a^*)$. Since SI syllables do not contribute to the linking number, an argument similar to the one from the proof of Lemma 3.8(2) can be used to conclude that the linker wins.

Subcase 2: Suppose the tangle word contains multiple strings of NSI syllables. Then we can apply the strategy from all but the last sentence of Subcase 1 of Case 1 to each tangle subword $(a_{k+1}, a_{k+2}, \dots, a_m, b_m^*)$, where $(a_{k+1}, a_{k+2}, \dots, a_m)$ is a nonfinal string of NSI syllables and where b_m^* is an SI syllable, so that each such subword can be reduced to $(0, 0)$. These subwords can then be removed from the tangle word by applying Proposition 2.16(2). If the final string of NSI syllables is not followed by a final SI syllable, then the tangle word can be reduced to (2)

or (-2) and the linker wins by Lemma 3.8. If the final string of NSI syllables is followed by a final SI syllable, then the argument from Subcase 1 can be applied in its entirety to conclude that the linker wins by Lemma 3.8.

We will now prove statement (2). Assume $(a_1, \dots, a_n)$ contains an odd number of SIs. Then the first player should begin by playing arbitrarily on an SI. This will effectively reduce the game to one with an even number of unresolved SIs where the first player is now playing second on this new link pseudodiagram. But then the first player can use the strategy from the proof of statement (1) to guarantee a win. $\square$

3C. The linking-unlinking game for general two-component link shadows. We now broaden our focus from rational two-component link shadows to general two-component link shadows. In particular, we will focus the majority of our attention on finding winning strategies for the linker. First, we introduce the notion of a partial linking number that can be applied to a link pseudodiagram throughout the linking-unlinking game.

Definition 3.13. The *pseudolinking number* of an oriented two-component link pseudodiagram is defined to be half of the sum of the crossing signs, where the sum is taken over all of the resolved non-self-intersections (NSIs) of the link pseudodiagram. Note that an oriented two-component link shadow has a pseudolinking number of 0 and note that the pseudolinking number of an oriented two-component link diagram is the linking number of the diagram.

In what follows, we use the pseudolinking number, the presence of NSIs, and the parity of the number of SIs to provide winning strategies for the linking-unlinking game played on a large family of general two-component link shadows. In particular, we present winning strategies for the unlinker in a very specific pathological case and for the linker in other cases.

Theorem 3.14. *Suppose we have a shadow of a two-component link:*

- (1) *If the shadow contains no NSIs, then the unlinker wins.*
- (2) *If the shadow contains a nonzero number of NSIs and an even number of SIs, then the linker has a winning strategy when playing second.*
- (3) *If the shadow contains a nonzero number of NSIs and an odd number of SIs, then the linker has a winning strategy when playing first.*

Proof. We begin by proving statement (1). Since the two-component link shadow contains no NSIs, the game starts on a split two-component link shadow and the unlinker wins automatically because resolving crossings will never create an unsplittable link diagram.

We now prove statement (2). Assign an arbitrary orientation to the two-component link shadow. If the unlinker plays on an SI, the linker should respond by also playing

on an SI. Since the number of SIs is even, the linker will always be able to respond on a remaining SI until all SIs are resolved. Note that the SIs do not affect the (pseudo)linking number, so how the linker responds is arbitrary.

Recall that the pseudolinking number of a two-component link shadow (or a two-component link pseudodiagram with only SIs resolved) is 0. Thus, after the unlinker plays for the first time on an NSI, the two-component link pseudodiagram will have a positive pseudolinking number of $\frac{1}{2}$ or a negative pseudolinking number of $-\frac{1}{2}$. The linker should then respond on any other NSI by resolving the crossing to have the same sign as the crossing resolved by the unlinker on the previous move, which will change the pseudolinking number to 1 or -1 . For the remaining NSIs, when the unlinker plays on an NSI, the linker should respond on a remaining NSI by resolving the crossing to have sign opposite to the crossing resolved by the unlinker on the previous move. Since Proposition 2.13 guarantees that every two-component link pseudodiagram contains an even number of NSIs, the linker will always be able to respond on a remaining NSI until all NSIs are resolved.

Notice that the linker's response strategy preserves the pseudolinking number at 1 or -1 (after their first response on an NSI) for the remainder of the game. Consequently, when the linker makes the final move, the resulting two-component link diagram will have a nonzero linking number. This implies that the link diagram is unsplittable and, therefore, the linker wins.

We now prove statement (3). Assign an arbitrary orientation to the two-component link shadow. The linker should begin by playing arbitrarily on an SI. This will effectively reduce the game to one with an even number of unresolved SIs where the linker is now playing second on this new link pseudodiagram. But then the linker can use the strategy from the proof of statement (2) to guarantee a win. $\square$

Conclusion

In Section 3 of this paper, we introduced a new two-player combinatorial game played on shadows of two-component link diagrams, called the linking-unlinking game, where players take turns resolving crossings. One player, the linker, wins if the resulting two-component link diagram is unsplittable, while the other player, the unlinker, wins if the resulting two-component link diagram is splittable.

By studying the decomposition of a rational tangle word into SIs and NSIs, by applying the R2- and anti-R2-strategies, and by utilizing tangle equivalences, we were able to find winning strategies for playing the linking-unlinking game on any shadow of a two-component rational tangle closure (see Theorems 3.9(1) and 3.12).

By studying the pseudolinking numbers of general two-component link shadows, we were able to find winning strategies for the unlinker in the pathological case when the shadow contains no NSIs and winning strategies for the linker in half of

the cases when the shadow contains NSIs (see Theorem 3.14). Recall that ensuring a linking number of 0 does not guarantee that a link diagram will be splittable. This suggests that linking number arguments may not be as useful when seeking winning strategies for the unlinker in the remaining half of the cases when the shadow contains NSIs. As such, these cases currently remain open.

Question 3.15. Can explicit winning strategies for playing the linking-unlinking game on two-component link shadows be found in the remaining cases not covered by Theorem 3.14?

If winning strategies for playing the linking-unlinking game on all two-component link shadows cannot be found, then it would be interesting to find families of two-component link shadows beyond rational tangle closures for which winning strategies can be found.

Question 3.16. For which new families of link shadows can explicit winning strategies for playing the linking-unlinking game be found?

References

- [Adams 1994] C. C. Adams, *The knot book: an elementary introduction to the mathematical theory of knots*, Freeman, New York, 1994. MR Zbl
- [Bollobás 1998] B. Bollobás, *Modern graph theory*, Graduate Texts in Math. **184**, Springer, 1998. MR
- [Brown et al. 2017] S. Brown, F. Cabrera, R. Evans, G. Gibbs, A. Henrich, and J. Kreinbihl, “The region unknotting game”, *Math. Mag.* **90**:5 (2017), 323–337. MR Zbl
- [Ganzell et al. 2014] S. Ganzell, A. Meadows, and J. Ross, “Twist untangle and related knot games”, *Integers* **14** (2014), art. id. G4. MR Zbl
- [Hanaki 2010] R. Hanaki, “Pseudo diagrams of knots, links and spatial graphs”, *Osaka J. Math.* **47**:3 (2010), 863–883. MR Zbl
- [Henle 1979] M. Henle, *A combinatorial introduction to topology*, Freeman, San Francisco, 1979. MR Zbl
- [Henrich et al. 2011] A. Henrich, N. MacNaughton, S. Narayan, O. Pechenik, R. Silversmith, and J. Townsend, “A midsummer knot’s dream”, *College Math. J.* **42**:2 (2011), 126–134. MR Zbl
- [Johnson 2011] W. Johnson, “The knotting-unknotting game played on sums of rational shadows”, preprint, 2011. arXiv
- [Johnson and Henrich 2017] I. Johnson and A. K. Henrich, *An interactive introduction to knot theory*, Dover, Mineola, NY, 2017. Zbl

Received: 2018-07-30 Revised: 2019-05-17 Accepted: 2019-06-11

agiambrone@elmira.edu

Elmira College, Elmira, NY, United States

jmurp61@lsu.edu

Louisiana State University, Baton Rouge, LA, United States

involve

msp.org/involve

INVOLVE YOUR STUDENTS IN RESEARCH

Involve showcases and encourages high-quality mathematical research involving students from all academic levels. The editorial board consists of mathematical scientists committed to nurturing student participation in research. Bridging the gap between the extremes of purely undergraduate research journals and mainstream research journals, *Involve* provides a venue to mathematicians wishing to encourage the creative involvement of students.

MANAGING EDITOR

Kenneth S. Berenhaut Wake Forest University, USA

BOARD OF EDITORS

Colin Adams	Williams College, USA	Robert B. Lund	Clemson University, USA
Arthur T. Benjamin	Harvey Mudd College, USA	Gaven J. Martin	Massey University, New Zealand
Martin Bohner	Missouri U of Science and Technology, USA	Mary Meyer	Colorado State University, USA
Amarjit S. Budhiraja	U of N Carolina, Chapel Hill, USA	Frank Morgan	Williams College, USA
Pietro Cerone	La Trobe University, Australia	Mohammad Sal Moslehian	Ferdowsi University of Mashhad, Iran
Scott Chapman	Sam Houston State University, USA	Zuhair Nashed	University of Central Florida, USA
Joshua N. Cooper	University of South Carolina, USA	Ken Ono	Univ. of Virginia, Charlottesville
Jem N. Corcoran	University of Colorado, USA	Yuval Peres	Microsoft Research, USA
Toka Diagana	Howard University, USA	Y.-F. S. Pétermann	Université de Genève, Switzerland
Michael Dorff	Brigham Young University, USA	Jonathon Peterson	Purdue University, USA
Sever S. Dragomir	Victoria University, Australia	Robert J. Plemmons	Wake Forest University, USA
Joel Foisy	SUNY Potsdam, USA	Carl B. Pomerance	Dartmouth College, USA
Errin W. Fulp	Wake Forest University, USA	Vadim Ponomarenko	San Diego State University, USA
Joseph Gallian	University of Minnesota Duluth, USA	Bjorn Poonen	UC Berkeley, USA
Stephan R. Garcia	Pomona College, USA	József H. Przytycki	George Washington University, USA
Anant Godbole	East Tennessee State University, USA	Richard Rebarber	University of Nebraska, USA
Ron Gould	Emory University, USA	Robert W. Robinson	University of Georgia, USA
Sat Gupta	U of North Carolina, Greensboro, USA	Javier Rojo	Oregon State University, USA
Jim Haglund	University of Pennsylvania, USA	Filip Saidak	U of North Carolina, Greensboro, USA
Johnny Henderson	Baylor University, USA	Hari Mohan Srivastava	University of Victoria, Canada
Glenn H. Hurlbert	Virginia Commonwealth University, USA	Andrew J. Sterge	Honorary Editor
Charles R. Johnson	College of William and Mary, USA	Ann Trenk	Wellesley College, USA
K. B. Kulasekera	Clemson University, USA	Ravi Vakil	Stanford University, USA
Gerry Ladas	University of Rhode Island, USA	Antonia Vecchio	Consiglio Nazionale delle Ricerche, Italy
David Larson	Texas A&M University, USA	John C. Wierman	Johns Hopkins University, USA
Suzanne Lenhart	University of Tennessee, USA	Michael E. Zieve	University of Michigan, USA
Chi-Kwong Li	College of William and Mary, USA		

PRODUCTION

Silvio Levy, Scientific Editor

Cover: Alex Scorpan

See inside back cover or msp.org/involve for submission instructions. The subscription price for 2019 is US \$195/year for the electronic version, and \$260/year (+\$35, if shipping outside the US) for print and electronic. Subscriptions, requests for back issues and changes of subscriber address should be sent to MSP.

Involve (ISSN 1444-4184 electronic, 1444-4176 printed) at Mathematical Sciences Publishers, 798 Evans Hall #3840, c/o University of California, Berkeley, CA 94720-3840, is published continuously online. Periodical rate postage paid at Berkeley, CA 94704, and additional mailing offices.

Involve peer review and production are managed by EditFlow® from Mathematical Sciences Publishers.

PUBLISHED BY

 **mathematical sciences publishers**
nonprofit scientific publishing

<http://msp.org/>

© 2019 Mathematical Sciences Publishers

involve

2019 vol. 12 no. 7

Asymptotic expansion of Warlimont functions on Wright semigroups	1081
MARCO ALDI AND HANQIU TAN	
A systematic development of Jeans' criterion with rotation for gravitational instabilities	1099
KOHL GILL, DAVID J. WOLLKIND AND BONNI J. DICHONE	
The linking-unlinking game	1109
ADAM GIAMBRONE AND JAKE MURPHY	
On generalizing happy numbers to fractional-base number systems	1143
ENRIQUE TREVIÑO AND MIKITA ZHYLINSKI	
On the Hadwiger number of Kneser graphs and their random subgraphs	1153
ARRAN HAMM AND KRISTEN MELTON	
A binary unrelated-question RRT model accounting for untruthful responding	1163
AMBER YOUNG, SAT GUPTA AND RYAN PARKS	
Toward a Nordhaus–Gaddum inequality for the number of dominating sets	1175
LAUREN KEOUGH AND DAVID SHANE	
On some obstructions of flag vector pairs (f_1, f_{04}) of 5-polytopes	1183
HYE BIN CHO AND JIN HONG KIM	
Benford's law beyond independence: tracking Benford behavior in copula models	1193
REBECCA F. DURST AND STEVEN J. MILLER	
Closed geodesics on doubled polygons	1219
IAN M. ADELSTEIN AND ADAM Y. W. FONG	
Sign pattern matrices that allow inertia $\mathbb{S}_n$	1229
ADAM H. BERLINER, DEREK DEBLIECK AND DEEPAK SHAH	
Some combinatorics from Zeckendorf representations	1241
TYLER BALL, RACHEL CHAISER, DEAN DUSTIN, TOM EDGAR AND PAUL LAGARDE	



1944-4176(2019)12:7;1-Z