CONTEXT DIRECTED REVERSALS AND THE CILIATE DECRYPTOME

C.L. JANSEN, M. SCHEEPERS, S.L. SIMON AND E. TATUM

ABSTRACT. Prior studies of the efficiency of the block interchange (swap) and the reversal sorting operations on (signed) permutations identified specialized versions of these operations. These specialized operations are here called context directed reversal, abbreviated \textit{cdr}, and context directed swap, abbreviated \textit{cds}. Prior works have also characterized which (signed) permutations are sortable by \textit{cdr} or by \textit{cds}.

It is now known that when a permutation \(\pi\) is \textit{cds} sortable in \(n\) steps, then any \(n\) consecutive applicable \textit{cds} operations will sort \(\pi\). Examples show that this is not the case for \textit{cdr}. This phenomenon is the focus of this paper. It is proven that if a signed permutation is \textit{cdr} sortable, then any \textit{cdr} fixed point of it is \textit{cds} sortable (the \textit{cds} Rescue Theorem). The \textit{cds} Rescue Theorem is discussed in the context of a mathematical model for ciliate micronuclear decryption.

It is also known that if applications of \textit{cds} to a permutation \(\pi\) reaches a \textit{cds} fixed point in \(n\) steps, then any \(n\) consecutive applicable \textit{cds} operations will terminate in a \textit{cds} fixed point of \(\pi\). This is not the case for \textit{cdr}: It is proven that though for a given signed permutation the number of \textit{cdr} operations leading to different \textit{cdr} fixed points may be different from each other, the parity of the number of operations is the same (the \textit{cdr} Parity Theorem). This result provides a solution to two previously formulated decision problems regarding certain combinatorial games.

1. Introduction

The scope, efficiency and robustness of sorting algorithms are of high interest since sorting is used to prepare data for application of various information processing algorithms. The mathematical study of permutation sorting correspondingly has a long history.

Applications of permutation sorting in biological studies have been stimulated by the discovery that often the positions of genes on the chromosomes of an organism A are a permutation of the positions of the corresponding genes on the chromosomes of another organism B. Dobzhansky and Sturtevant [18] proposed using the minimum number of reversals required to sort the gene order of organism A to that of organism B as a measure of the evolutionary distance between A and B. Hannenhalli and Pevzner [12] found an efficient algorithm for determining the minimum number of reversals when a signed permutation describes this relation between genes in A and in B. That work identified a particular class of reversals, named oriented reversals, as instrumental to finding that minimum number. Other advances in the study of (oriented) reversals may be found in [2, 3, 19]. In an independent line of investigation this specific type of reversal has been postulated as one of the two sorting operations executed in ciliates during the process of converting a micronuclear precursor of a
gene to its functional form. In [8] and other ciliate literature this special reversal operation has been denoted hi, an abbreviation for hairpin inverted repeat. We shall call this special reversal operation context directed reversal, abbreviated cdr.

Also block interchanges, and the special case of transpositions, have been studied as permutation sorting operations. Christie [5] studied the problem of finding the minimum number of block interchanges required to sort one permutation to another. A certain constrained class of block interchanges, called minimal block interchanges in [5], emerged as instrumental for efficient sorting by block interchanges. Again, independently, the model for ciliate micronuclear decryption postulates a constrained block interchange operation as a sorting operation towards accomplishing decryption of micronuclear precursors of genes. In ciliate literature, see for example [8], these constrained block interchanges are called dlad operations. Some instances of the minimal block interchanges identified by Christie are indeed dlad operations. We shall refer to a dlad operation as a context directed swap, abbreviated cds.

The cds Inevitability Theorem, a result from [1], shows that if a permutation is cds sortable, then in fact indiscriminate applications of cds will successfully sort the permutation. The advantage for the ciliate decryptome is that such a sorting operation does not require additional resources to direct strategic choices for successful sorting by cds. However, the cdr sorting operation does not have this inevitability feature: Indiscriminately applying cdr sorting operations to a cdr sortable (or reverse cdr sortable) signed permutation may terminate in an unsuccessful sorting. This fact motivates several questions, including: (1) Do signed permutations that are cdr sortable, but not indiscriminately cdr sortable, actually occur in ciliates? (2) If “yes”, how does the ciliate decryptome succeed in sorting such signed permutations? and (3) Does counting the number of cdr operations required to sort a signed permutation π entail finding a successful sequence of cdr operations that sort π?

Regarding Question (1): [7] reports the DNA sequences for the micronuclear precursors of the Actin I gene for the ciliate species Uroleptus pisces 1 and Uroleptus pisces 2. The first is available from [10] under Accession Number AF508053.1, and is representable by the signed permutation \([1, 3, -7, -5, 14, 2, 4, 6, 9, 12, -11, -8, 13, 15, -10]\). The second is available from [10] under Accession Number AY373659.1, and is representable by the signed permutation \([2, 4, 6, 9, 12, -11, -8, 13, 15, -10, 1, 3, -7, -5, 14]\). Both are cdr sortable, yet some sequences of applications of cdr result in a cdr fixed point different from the identity. This fact suggests that unless the applications of cdr operations during micronuclear decryption follow a strategy not yet discovered in the laboratory, cdr fixed points other than the identity would be encountered results of the decryption process.

Regarding Question (2): We prove the perhaps surprising result, Theorem 9.5, named the cds Rescue Theorem, that any cdr fixed point of a cdr sortable signed permutation is a cds sortable permutation. Thus, the cds Inevitability Theorem of [1] implies that the cds sorting operation plays a rescue role in the ciliate decryptome.

Regarding Question (3): In prior work [2,12,19], finding the minimum number of reversals to sort a permutation included finding a strategic selection of cdr operations that sort a cdr sortable signed permutation. The cds Inevitability Theorem plus the cds Rescue Theorem provide another approach. The cdr Steps Theorem, Theorem 9.6, determines the number of cdr applications required to sort a cdr sortable signed permutation by indiscriminate applications of cdr and cds.

Different cdr fixed points of a given signed permutation are not necessarily reached in the same number of cdr sorting steps. But the parity of the number of cdr operations...
required to reach a \textit{cdr} fixed point is an invariant of each signed permutation - the \textit{cdr} Parity Theorem, Theorem \[10.1\]. This result implies a linear time solution for the \textit{cdr} Misere and the \textit{cdr} Normal Play decision problems for combinatorial games featuring \textit{cdr}, defined previously in \[1\].

Our paper is organized as follows. After introducing basic terminology and the theoretical background information from prior work we prove the \textit{cds} Rescue Theorem and the \textit{cdr} Steps Theorem in Section 9, and the \textit{cdr} Parity Theorem in Section 10. Finally we discuss our results in the context of the ciliate micronuclear decryption model proposed in \[15\] [16].

For readers interested in the biological connections of this work we recommend the two textbooks \[8\] and \[9\].

2. Notation and Terminology

We consider both signed permutations and unsigned permutations. An unsigned permutation will simply be called a permutation. In either case we consider these as one-to-one and onto functions from the appropriate finite set to itself. For a permutation $\pi$ the notation

$$\pi = [a_1, \ldots, a_n]$$

denotes that the permutation $\pi$ maps $a_i$ to $i$. For a signed permutation $\pi$ the notation in (1) denotes that the signed permutation $\pi$ maps $a_i$ to $i$ and $-a_i$ to $-i$. The symbol $S_n$ denotes the set of permutations as in (1), while $S_n^\pm$ denotes the set of such signed permutations.

The definitions of the context directed sorting operations use the notion of a pointer: Consider a (signed) permutation $\alpha = [p_1, p_2, \ldots, p_i, \ldots, p_j, \ldots, p_n]$. Consider the entry $p_i$ of $\alpha$ and say $p_i$ is the integer $k$. Then the head pointer, or simply head of $p_i$ is the ordered pair $(k, k+1)$, while the tail pointer, or simply tail, of $p_i$ is the ordered pair $(|k|, |k|+1)$.

In displays where pointer locations are emphasized we will use notation as follows

$$\alpha = \begin{cases} [p_1, p_2, \ldots, (k-1, k) p_i (k, k+1), \ldots, p_j, \ldots, p_n] & \text{or alternately} \\ [p_1, p_2, \ldots, t p_i^h, \ldots, p_j, \ldots, p_n] & \text{when } p_i = |p_i| = k, \text{ and} \\ [p_1, p_2, \ldots, (k, k+1) p_i (k-1, k), \ldots, p_j, \ldots, p_n] & \text{or alternately} \\ [p_1, p_2, \ldots, h p_i^t, \ldots, p_j, \ldots, p_n] & \text{when } -p_i = |p_i| = k. \end{cases}$$

3. The oriented overlap graph of a signed permutation

Consider a signed permutation $\alpha \in S_n^\pm$. For each pointer $(i, i+1)$ appearing in $\alpha$, draw an arc between the head and the tail occurrence of this pointer. One possible scenario is depicted in Figure \[1\].

\begin{figure}[ht]
\centering
\includegraphics[width=\textwidth]{figure1}
\caption{An arc between the head and tail occurrences of a pointer in $\alpha$}
\end{figure}

The overlap graph $O(\alpha) = (V, E)$ of signed permutation $\alpha$ has

- vertex set $V$, the set of pointers of $\alpha$, and
• edge set $\mathcal{E}$, the set of pairs \{p, q\} of vertices p and q for which the arcs associated with p and q have nonempty intersection.

Following [12, 19] we designate a vertex of $\mathcal{O}(\alpha)$ as oriented if the corresponding arc is between pointers of entries of opposite sign in $\alpha$. In figures oriented vertices are denoted by filled circles, while unoriented vertices are denoted by unfilled circles.

**Example 3.1.** Constructing the overlap graph of the signed permutation $T = [1, -5, -2, 4, -3, 6]$.

![Figure 2. Construction of the overlap graph for T.](image)

Let $G = (V, E)$ be a graph. Declare vertices $x$ and $y$ of $G$ in reach if there are vertices $x_0, x_1, \ldots, x_m$ such that $x = x_0$ and $y = x_m$, and for each $i < m$ we have \{\{x_i, x_{i+1}\}\} \in \mathcal{E}$. The “in reach” relation is an equivalence relation on the vertex set of $G$. If an equivalence class of this relation has more than one element it is called a component of $G$. If this equivalence relation has only one equivalence class, we say that $G$ is a connected graph. If $(G, f)$ is an oriented graph, then a component of $G$ is said to be an oriented component if some vertex belonging to the component is oriented. The member of an equivalence class with only one element is said to be an isolated vertex.

**Example 3.2.** The signed permutation $S = [-6, 3, -4, 2, 5, -1, 7, 9, 8, 10]$. The overlap graph of $S$ follows:

![Figure 3. Overlap graph for S.](image)

The oriented graph in Figure 3 has an isolated vertex (that is oriented), an unoriented component, and an oriented component.

4. **Context directed reversals**

Let a signed permutation $\alpha$ with entries $x$ and $y$ as well as a pointer $(i, i+1)$ be given. Suppose that the entries $x$ and $y$ of $\alpha$ each is associated with the pointer $(i, i+1)$\(^1\). We define $\beta = \text{cdr}_{(i,i+1)}(\alpha)$ by cases:

\[^1\text{Thus, } (i, i+1) \text{ is the head pointer of one, and the tail pointer of the other, and } \{i, i+1\} = \{|x|, |y|\}.\]
Case 1: $\alpha = [\cdots, (i, i+1)x, \cdots, y_{(i, i+1)}, \cdots]$ or $\alpha = [\cdots, x_{(i, i+1)}, \cdots, (i, i+1)y, \cdots]$

In this case $x \cdot y = 0$, and we define $\beta = \alpha$.

Case 2: $\alpha = [\cdots, a, (i, i+1)x, \cdots, b, (i, i+1)y, \cdots]$.

In this case $\beta$ is obtained from $\alpha$ by reversing the order of all entries starting at $a$ and ending at $b$, and changing the signs of these entries. Thus,

$$\beta = \text{cdr}_{(i, i+1)}(\alpha) = [\cdots, a, -b, \cdots, -x_{(i, i+1)}, (i, i+1)y, \cdots]$$

Case 3: $\alpha = [\cdots, x_{(i, i+1)}, a, \cdots, y_{(i, i+1)}, b, \cdots]$.

In this case $\beta$ is obtained from $\alpha$ by reversing the order of all entries starting at $a$ and ending at $b$, and changing the signs of these entries. Thus,

$$\beta = \text{cdr}_{(i, i+1)}(\alpha) = [\cdots, x_{(i, i+1)}, (i, i+1)-y, \cdots, -a, b, \cdots]$$

**Example 4.1.** In the signed permutation $\pi = [-2, 1, -4, 3]$ the pointer $(2, 3)$ is the head pointer of $-2$ and the tail pointer of $3$, entries with opposite sign. Thus the context directed reversal $\text{cdr}_{(2, 3)}$ is applicable to $\pi$, and $\text{cdr}_{(2, 3)}(\pi) = [4, -1, 2, 3]$.

Context directed reversals have been called oriented reversals in\cite{12,19}, and hi, the hairpin inversion in\cite{8} and related literature. A fundamental theorem from\cite{12} implies

**Theorem 4.2** (Hannenhalli-Pevzner). *If the overlap graph of the signed permutation $\alpha$ has no unoriented components, then $\alpha$ is sortable by applications of cdr.*

Another fundamental theorem from\cite{12} implies (see the remarks around\cite{19} Theorem 2):

**Theorem 4.3** (Hannenhalli-Pevzner). *If the signed permutation $\alpha$ is sortable by applications of cdr, then all sequences of successive cdr operations that sort $\alpha$ are of the same length.*

5. LOCAL COMPLEMENTATION OF ORIENTED GRAPHS, AND THE OPERATION gcdr

The operation cdr on a signed permutation can be simulated by a corresponding operation on the overlap graph of the signed permutation. This operation, introduced in the prior work\cite{2,12,19}, is described next.

For a graph $G = (V, E)$ and a vertex two-coloring $f : V \rightarrow \{0, 1\}$ the pair $(G, f)$ is said to be an oriented graph. Vertices $v$ with $f(v) = 0$ are said to be oriented vertices, while the vertices $v$ with $f(v) = 1$ are called unoriented vertices. For a given set $S$ and an oriented graph $(G, f)$ we define the oriented graph

$$(G', f') = \text{lc}((G, f), S)$$

as follows:

1. $G' = (V', E')$, where $V' = V$ and $E' = (E \setminus [S]^2) \cup ([V \cap S]^2 \setminus E)$.
2. $f' : V' \rightarrow \{0, 1\}$ is defined by

$$f'(v) = \begin{cases} 1 - f(v) & \text{if } v \in S \\ f(v) & \text{otherwise} \end{cases}$$

The oriented graph $(G', f')$ is said to be the $S$-local complement of oriented graph $(G, f)$. It is evident that when $S \cap V = \emptyset$, then $(G, f) = \text{lc}((G, f), S)$. Thus we may assume that $S \subseteq V$.

An example of an oriented graph on seven vertices is given in Figure 4a. The resulting graph $\text{lc}((G, f), S)$ is displayed in Figure 4b.

\footnote{In other words, $x$ and $y$ have the same sign.}
Let \((G, f)\) be an oriented graph. Consider a vertex \(p\) of \(G\). If \(p\) is not an oriented vertex of \(G\), then declare \((G', f') = (G, f)\). However, if \(p\) is an oriented vertex of \(G\), then define:

\[
N(p) = \{q : \{p, q\} \text{ is an edge of } G\} \cup \{p\}.
\]

Finally, define

\[
\text{gcdr}((G, f), p) = \text{lc}((G, f), N(p)).
\]

In [19] the authors use the notation \(G/p\) for \(\text{gcdr}((G, f), p)\). Observe that if in \(G\) the vertex \(p\) is oriented, then in \(\text{gcdr}((G, f), p)\) the vertex \(p\) is unoriented and isolated.

### 6. Simulating \(\text{cdr}\) on the oriented overlap graph of a signed permutation

We next relate an application of the \(\text{cdr}\) sorting operation on a signed permutation \(\alpha\) to an application of \(\text{gcdr}\) on the oriented overlap graph of \(\alpha\).

**Example 6.1.** The signed permutation \(\alpha = [3, -8, -2, 5, 1, -7, 4, 6]\).

\[
\begin{align*}
\text{(A) The overlap graph } (G, f) \text{ of } \alpha. & \quad \text{(B) Overlap graph of } \text{cdr}_{(6,7)}(\alpha). \\
(1,2) & \quad (2,3) \\
(7,8) & \quad (3,4) \\
(6,7) & \quad (4,5) \\
(5,6) & \quad (5,6)
\end{align*}
\]

\(\Xi_n : S^+_n \rightarrow M_n\) denotes the map that associates with each signed permutation \(\alpha \in S^+_n\) its corresponding oriented overlap graph. The following theorem has been proven in [12], and it shows that the diagram in Figure 6 commutes:

**Theorem 6.2.** Let \(n\) be a positive integer and let \(\alpha\) be a signed permutation in \(S^+_n\) and let \(p\) be a pointer of \(\alpha\). Then \(\Xi_n(\text{cdr}_p(\alpha)) = \text{gcdr}(\Xi_n(\alpha), p)\).
A good exposition of this result is given in Section 2 of [19].

\[ \alpha \xrightarrow{\text{cdr}_p} \alpha' \]

\[ \Xi_n(\alpha) \xrightarrow{\text{gcdr}(\cdot, p)} \Xi_n(\alpha') \]

\textbf{Figure 6}

7. Oriented Components of graphs and \text{gcdr}.

The following result, Theorem 1 of [19], generalizes Theorem 4 of [12] from the context of oriented graphs arising from signed permutations, to the general context of oriented graphs.

\textbf{Theorem 7.1 (Fundamental Theorem of Oriented Graphs).} If each component of the oriented graph \((G, f)\) is oriented, then there exists an oriented vertex \(v\) of \(G\) such that each component of \(\text{gcdr}((G, f), v)\) is oriented.

Let \((G, f)\) be a finite oriented graph, and let \((v_1, \ldots, v_k)\) be a sequence of vertices of \(G\). Define \((G_1, f_1) = \text{gcdr}((G, f), v_1)\) and for \(i < k\), \((G_{i+1}, f_{i+1}) = \text{gcdr}((G_i, f_i), v_{i+1})\). Following [19] we say that \((v_1, \ldots, v_k)\) is a sequence of oriented vertices if \(v_1\) is an oriented vertex of \((G, f)\) and for each \(i < k\), \(v_{i+1}\) is an oriented vertex of \((G_i, f_i)\). An oriented sequence of vertices is a \text{maximal sequence} if \((G_k, f_k)\) has no oriented vertices. It is a \text{total sequence} if \((G_k, f_k)\) has only isolated, unoriented, vertices.

Observe that a total sequence is a maximal sequence, but a maximal sequence need not be a total sequence. Theorem 7.1 implies

\textbf{Corollary 7.2.} If each component of the oriented finite graph \((G, f)\) is oriented, then there is a total sequence of oriented vertices in \(G\).

\textbf{Problem 7.3.} Let \((G, f)\) be a connected oriented graph with an oriented vertex. Are all total sequences of oriented vertices for \(G\) of the same length?

8. The \text{cdr} Revision Theorem.

We now reformulate Theorem 3 from [19], named here the \text{cdr} Revision Theorem, for our purposes. Throughout this section we assume that \((G, f)\) is an oriented graph. We also allow for the presence of a number of isolated vertices. The vertex set of \(G\) is \(V\), while the edge set is \(E\).

\textbf{Theorem 8.1 (cdr Revision Theorem, [19]).} Let \((G, f)\) be a graph which has a unique component. Assume that this component is oriented. If \((v_1, v_2, \ldots, v_{m-1}, v_m)\) is a maximal, but not total, sequence of oriented vertices for the graph \((G, f)\), then there exists an \(\ell < m\) and a sequence \((r_1, r_2)\) of vertices of \(G\) such that:

\(1\) \((v_1, \ldots, v_\ell, r_1, r_2, v_{\ell+1}, \ldots, v_m)\) is a maximal oriented sequence in \((G, f)\), and
(2) for each \( j \) such that \( \ell < j \leq m \) neither \( r_1 \), nor \( r_2 \), is an oriented vertex in any of the graphs \( \gcdr(\gcdr(\cdots \gcdr((G, f), v_1), \cdots) v_{j-1}), v_j) \).

Repeated applications of the \( \text{cdr} \) Revision Theorem then leads to

**Corollary 8.2.** Let \((G, f)\) be a graph that has a unique component. If this component is oriented and \((v_1, v_2, \cdots, v_{m-1}, v_m)\) is a maximal, but not total, sequence of oriented vertices for \((G, f)\), then there exists an \( \ell < m \) and a sequence \((r_1, r_2, \cdots, r_{2k-1}, r_{2k})\) of vertices of \( G \) such that \((v_1, \cdots, v_\ell, r_1, r_2, \cdots, r_{2k-1}, r_{2k}, v_{\ell+1}, \cdots, v_m)\) is a total oriented sequence in \((G, f)\).

**Example 8.3.** Consider the signed permutation \( \pi = [1, 3, 5, -2, -6, 4] \). The oriented sequence \((5, 6))\) is maximal. Each two-term sequence in the first column of Figure 8 is an oriented sequence in the overlap graph of \( \pi \), while the corresponding three term sequence in the second column is maximal. By Theorem 8.1 for each of these maximal three-term

\[
\begin{array}{c|c}
((1,2), (2,3)) & ((1,2), (2,3), (5,6)) \\
((1,2), (3,4)) & ((1,2), (3,4), (5,6)) \\
((1,2), (4,5)) & ((1,2), (4,5), (5,6)) \\
((2,3), (1,2)) & ((2,3), (1,2), (5,6)) \\
((2,3), (3,4)) & ((2,3), (3,4), (5,6)) \\
((2,3), (4,5)) & ((2,3), (4,5), (5,6)) \\
\end{array}
\]

**Figure 8.** Oriented, and corresponding maximal sequences for \( \pi \).

sequences there is a pair of vertices of the overlap graph that can be inserted right before the term \((5, 6)\) to produce a five-term maximal oriented sequence. One example of such a five-term maximal sequence, \(( (1,2), (3,4), (4,5), (2,3), (5,6)) \), is a total sequence for the overlap graph of \( \pi \). By Theorem 4.3 all total sequences for the overlap graph of \( \pi \) are of length 5.

9. **The \( \text{cds} \) Rescue Theorem.**

As can be gleaned from Example 8.3, even when a signed permutation is \( \text{cdr} \)-sortable, not all applications of \( \text{cdr} \)-sorting operations successfully sort the permutation. A failed sorting during micronuclear decryption in a ciliate could have devastating consequences for the organism. In this section we prove that the hypothesized presence of also the \( \text{cds} \) sorting operation is highly advantageous for accomplishing successful sorting of signed permutations in the organism.
For the reader’s convenience we review information regarding cds. Suppose \( \alpha \) has a pair of pointers \( p = (x, x+1), q = (y, y+1) \). Then \( \beta = \text{cds}_{p,q}(\alpha) \), the result of the sorting operation is as follows:

**Case 1:** If the pointers do not appear in the order \( p \ldots q \ldots p \ldots q \) or in the order \( q \ldots p \ldots q \ldots p \) we define \( \beta = \text{cds}_{p,q}(\alpha) = \alpha \).

**Case 2:** If \( \alpha = \cdots, a, p x + 1, b, \ldots, c, y q, d, \ldots, e, x p, f, \ldots, g, q y + 1, h, \cdots \), define

\[
\beta = \text{cds}_{p,q}(\alpha) = \cdots, a, f, \ldots, g, d, \ldots, e, x p, p x + 1, b, \ldots, c, y q, q y + 1, h, \cdots
\]

**Case 3:** If \( \alpha = \cdots, a, x p, b, \ldots, c, y q, d, \ldots, e, p x + 1, f, \ldots, g, q y + 1, h, \cdots \), define

\[
\beta = \text{cds}_{p,q}(\alpha) = \cdots, a, x p, p x + 1, f, \ldots, g, d, \ldots, e, b, \ldots, c, y q, q y + 1, h, \cdots
\]

**Case 4:** If \( \alpha = \cdots, a, x p, b, \ldots, c, q y + 1, d, \ldots, e, p x + 1, f, \ldots, g, y q, h, \cdots \), define

\[
\beta = \text{cds}_{p,q}(\alpha) = \cdots, a, x p, p x + 1, f, \ldots, g, y q, q y + 1, d, \ldots, e, b, \ldots, c, h, \cdots
\]

**Case 5:** If \( \alpha = \cdots, a, p x + 1, b, \ldots, c, q y + 1, d, \ldots, e, x p, f, \ldots, g, y q, h, \cdots \), define

\[
\beta = \text{cds}_{p,q}(\alpha) = \cdots, a, f, \ldots, g, y q, q y + 1, d, \ldots, e, x p, p x + 1, b, \ldots, c, h, \cdots
\]

**Example 9.1.** We perform \( \text{cds}_{(3,4),(6,7)} \) on \( \alpha = [3, 6, 5, 2, 4, 8, 1, 7] \): Label \( \alpha \) with the relevant pointers: \( \alpha = [3,(3,4),6,(6,7),5,2,(3,4)4,8,1,(6,7),7] \). Performing the swap yields \( \beta = [3,4,8,1,5,2,6,7] \). Both 3, 4 and 6, 7 are now adjacent in \( \beta \).

**Lemma 9.2** (The cdr-cds Lemma). Let \( a, b, \) and \( c \) be pointers in a signed permutation \( \pi \). Assume that \( \beta = \text{cdr}_c \circ \text{cdr}_b \circ \text{cdr}_a(\pi) \). Also assume that neither \( \text{cdr}_a \) nor \( \text{cdr}_b \) applies to \( \gamma = \text{cdr}_c(\pi) \). Then \( \text{cds}_{(a,b)} \) applies to \( \gamma \) and \( \beta = \text{cds}_{(a,b)}(\gamma) \).

**Proof.** By hypothesis neither \( \text{cdr}_a \) nor \( \text{cdr}_b \) applies to \( \gamma = \text{cdr}_c(\pi) \). Thus in \( \gamma \) all occurrences of the pointer \( a \) are of the same sign, and similarly for the pointer \( b \). As \( \text{cdr}_a \) applies to the signed permutation \( \pi \), the two occurrences of pointer \( a \) in \( \pi \) have opposite signs. Thus in \( \pi \) one of these occurrences, and not both, of \( a \) is flanked by pointer \( c \), as depicted in Figure 9.

**Figure 9.** Pointers \( a \) and \( c \) in signed permutation \( \pi \).

Figure 10 depicts one of eight configurations regarding placement of and signs on \( a \) and \( c \). We shall give the argument for this depiction, leaving the similar arguments for the other cases to the reader. Now consider the possible placements of the pointer \( b \). There are three cases, depicted in Figure 10. We explain the first case, leaving the other two to the reader.

**Figure 10.** Three general cases for placement of pointers \( a, b \) and \( c \) in \( \pi \).
Case 1 of Figure 10: We write $\pi$ as the concatenation $$\pi = \pi_1 \pi_2 \pi_3 \pi_4 \pi_5 \pi_6 \pi_7$$
where: $\pi_1$ is the segment up to and including pointer $a$, $\pi_2$ is the segment from pointer $a$ up to and including pointer $b$, $\pi_3$ is the segment from pointer $-b$ up to and including pointer $-c$, $\pi_4$ is the segment from pointer $-c$ up to and including pointer $b$, $\pi_5$ is the segment from pointer $b$ up to and including pointer $-a$, $\pi_6$ is the segment from pointer $-a$ up to and including pointer $c$, and $\pi_7$ is the final segment.

Applying $\text{cdr}_a$ produces $\pi_1 - \pi_5 - \pi_4 - \pi_3 - \pi_2 \pi_6 \pi_7$. Then applying $\text{cdr}_b$ produces $\pi_1 - \pi_5 \pi_3 \pi_4 - \pi_2 \pi_6 \pi_7$. Finally applying $\text{cdr}_c$ produces $$\beta = \pi_1 - \pi_5 \pi_3 - \pi_6 \pi_2 - \pi_4 \pi_7.$$ 
On the other hand, first applying $\text{cdr}_c$ to $\pi$ produces $$\gamma = \pi_1 \pi_2 \pi_3 - \pi_6 - \pi_5 - \pi_4 \pi_7.$$ 
By tracking the movement of the pointers $a$ and $b$ and their signs in this step, we see that the segments $\pi_2$ and $\pi_5$ are exchanged when $\text{cds}_{a,b}$ is applied to $\gamma$, producing $\beta$.

Cases 2 and 3 of Figure 10 are treated similarly: Partition $\pi$ into the seven corresponding segments, and track the movement of the pointers as the sorting operations are applied. □

Example 8.3 continued. Application of $\text{cdr}_{(5,6)}$ to $\pi = [1, 3, 5, -2, -6, 4]$ produces $\gamma = [1, 3, 5, 6, 2, 4]$, to which no further context directed reversals can be applied. This corresponds to $((5,6))$ being a maximal oriented sequence in the overlap graph of $\pi$. The three-term oriented sequence $((1,2), (2,3), (5,6))$ is also maximal, and applying $\text{cdr}$ for these pointers in the order listed in the sequence produces $\delta = [1, 2, 3, 5, 6, 4]$. Observe that $\text{cds}_{(1,2),(2,3)}$ is applicable to $\gamma$, and that $\delta = \text{cds}_{(1,2),(2,3)}(\gamma)$.

Lemma 9.2 combined with the $\text{cdr}$ Revision Theorem, Theorem 8.1, and its corollary, leads to the $\text{cds}$ Rescue Theorem. In preparation for the proof of this theorem we first give the following two lemmas:

Lemma 9.3. Let $w \in V$ be an oriented vertex of $(\mathcal{G}, f)$. The following are equivalent:

(1) $(w)$ is a maximal oriented sequence.
(2) A vertex $v$ of $(\mathcal{G}, f)$ is oriented if, and only if, $\{w, v\}$ is an edge of $(\mathcal{G}, f)$.

Proof. (1) $\Rightarrow$ (2): Assume that $(w)$ is a maximal oriented sequence. Consider any $v \notin N(w)$. Then the orientation of $v$ in the graph $(\mathcal{G}_1, f_1) = \text{gcdr}((\mathcal{G}, f), w)$ is unchanged from its orientation in $(\mathcal{G}, f)$. As $(w)$ is a maximal sequence of oriented vertices, it follows that $v$ is unoriented in $(\mathcal{G}_1, f_1)$, and thus in $(\mathcal{G}, f)$. For the rest of (2), we assume that vertex $v$ is not oriented in $(\mathcal{G}, f)$. If $v$ were an element of $N(w)$, then in $(\mathcal{G}_1, f_1)$ $v$ would be an oriented vertex. But this contradicts the maximality of $(w)$.

(2) $\Rightarrow$ (1): Assume that, contrary to (2), there is an oriented vertex $v$ of $(\mathcal{G}, f)$ such that $\{v, w\}$ is not an edge of $\mathcal{G}$. Then $v$ is still an oriented vertex in $(\mathcal{G}_1, f_1)$, whence the oriented sequence $(w)$ is not maximal. □

It follows that if $(w)$ is a maximal sequence in an oriented graph, then no sequence of the form $(a, w)$ of vertices of the oriented graph can be an oriented sequence. Moreover, if an oriented graph has a maximal sequence $(w)$, then only one of its components is oriented.
Lemma 9.4. Let \( w \) be a vertex in the oriented graph \((G, f)\). If \((w)\) is a maximal sequence, then any oriented sequence of the form \((u, v, w)\) of \((G, f)\) is maximal.

Proof. Assume that there is an oriented sequence \((u, v, w)\) that is not maximal. Choose a vertex \(x\) such that \((u, v, w, x)\) is an oriented sequence. Put \((G_1, f_1) = \gcd((G, f), u), (G_2, f_2) = \gcd((G_1, f_1), v), (G_3, f_3) = \gcd((G_2, f_2), w)\). As \(u\) is an oriented vertex of \((G, f)\) (by hypothesis) Lemma 9.3 implies:

\[
(2) \quad w \in N(u) \text{ in } (G, f).
\]

But then \(w\) is unoriented in \((G_1, f_1)\), while it is oriented in \((G_2, f_2)\), implying

\[
(3) \quad w \in N(v) \text{ in } (G_1, f_1).
\]

Suppose that \(v \notin N(u)\) in \((G, f)\). Then as the orientation of \(v\) in \((G_1, f_1)\) is the same as its orientation in \((G, f)\), we find that \(v\) is an oriented vertex of \((G, f)\) and thus in \(N(w)\), by Lemma 9.3. If, on the other hand, \(v \in N(u)\) in \((G, f)\), then as the orientation of \(v\) in \((G_1, f_1)\) is opposite to its orientation in \((G, f)\), we find that \(v\) is an unoriented vertex of \((G, f)\) and thus not a member of \(N(w)\), by Lemma 9.3. Thus, we must consider the following two cases, (A) and (B):

\[(A) \quad v \notin N(u) \quad \quad (B) \quad v \in N(u)\]

Case (A) \(v \notin N(u)\): There are two subcases regarding \(x\):

Case 1: \(x \in N(w)\). By Lemma 9.3 \(x\) is an oriented vertex in \((G, f)\). We consider the four possibilities \(x \notin N(u) \cup N(v), x \in N(u) \setminus N(v), x \in N(v) \setminus N(u)\) and \(x \in N(v) \cap N(u)\).

1. \(x \notin N(u) \cup N(v)\): In this case \(x\) is oriented in \((G_2, f_2)\) and still has an edge with \(w\), so that \(x\) is unoriented in \((G_3, f_3)\).
2. \(x \in N(u) \setminus N(v)\): Now \(u, w\) and \(x\) are oriented and pairwise have an edge, so that in \((G_1, f_1)\) these three vertices are unoriented and \(x\) has no edge with any of \(u, v\) or \(w\). But then \(x\) is unoriented also in \((G_3, f_3)\).
3. \(x \in N(v) \setminus N(u)\): In this case in \((G_1, f_1)\) the vertices \(x, v\) and \(w\) pairwise have edges, \(w\) is unoriented and \(v\) and \(x\) are oriented. But then in \((G_2, f_2)\) there is no edge among any two of the vertices \(x, v\) and \(w\), and \(x\) is unoriented. It follows that \(x\) is unoriented also in \((G_3, f_3)\).
4. \(x \in N(u) \cap N(v)\): In this case the sets of vertices \(\{u, x, w\}\) and \(\{v, x, w\}\) each pairwise have edges in \((G, f)\), and all are oriented vertices. Then in \((G_1, f_1)\) the edges \(\{x, v\}\) and \(\{w, v\}\) are still present, but \(x\) and \(w\) are both unoriented while \(v\) is oriented. Thus in \((G_2, f_2)\) \(w\) and \(x\) are oriented and there is an edge \(\{w, x\}\). But then \(x\) is unoriented in \((G_3, f_3)\).

Case 2: \(x \notin N(w)\) Similarly, a consideration of cases shows that also in this case we would have a contradiction with the assumption that \((u, v, w, x)\) is an oriented sequence.

1. \(x \notin N(u) \cup N(v)\): In this case \(x\) is unoriented in \((G_3, f_3)\).
Then in (\(G_1, f_1\)) and a member of \(N(w)\) of the now unoriented \(w\). As we are in case (A), also \(v \in N(w)\) and \(v\) is oriented. But then in \((G_2, f_2)\) both \(w\) and \(x\) are oriented and \(\{w, x\}\) is an edge in \((G_2, f_2)\). It follows that \(x\) is unoriented in \((G_3, f_3)\).

(3) \(x \in N(v) \setminus N(u)\): Now in \((G_1, f_1)\) \(x\) is still a member of \(N(v)\) and unoriented, so that in \((G_2, f_2)\) both \(x\) and \(w\) are oriented and \(\{w, x\}\) is an edge. But then \(x\) is unoriented in \((G_3, f_3)\).

(4) \(x \in N(u) \cap N(v)\): In \((G_1, f_1)\) \(x\) and \(v\) are oriented while \(w\) is unoriented, and there is an edge between any two of \(\{w, x, v\}\). But then in \((G_2, f_2)\) \(x\) is unoriented, and there is no edge between \(w\) and \(x\). It follows that in \((G_3, f_3)\), \(x\) is unoriented.

This analysis shows that when \(v\) is not in \(N(u)\) in \((G, f)\), then \((u, v, w, x)\) is not an oriented sequence.

Also in Case (B) a similar case analysis exhibits a contradiction with the assumption that \((u, v, w, x)\) is an oriented sequence.

Thus, existence of a non-maximal oriented sequence \((u, v, w)\) leads to a contradiction. \(\square\)

**Theorem 9.5** (The \textbf{cds} Rescue Theorem). Let \(\pi\) be a \textbf{cdr}-sortable signed permutation. Then all \textbf{cdr} fixed points of \(\pi\) are \textbf{cds}-sortable.

**Proof.** Let \(\pi\) be a \textbf{cdr} sortable signed permutation. Then the overlap graph of \(\pi\) has no unoriented components. We prove this result by induction on the positive integer \(\pi\) which is a member of \(S^+_n\). For \(n = 1\) there is nothing to prove.

Thus, assume \(n > 1\) and that the statement of the theorem has been confirmed for all \(k < n\). Let \(\pi \in S^+_n\) be a \textbf{cdr}-sortable permutation with no adjacencies, and with some (non-sorted) \textbf{cdr}-fixed point \(\beta\). Consider an oriented sequence \((a_1, \ldots, a_m)\) of pointers used successively to obtain \(\beta\) from \(\pi\).

**Case 1: \(m = 1\).** Then \((a_1)\) is a maximal oriented sequence for the oriented overlap graph \((G, f)\) associated with \(\pi\). Now by the Fundamental Theorem of Oriented Graphs, Theorem 7.1, choose an oriented vertex \(r_1\) of \(\pi\) such that the overlap graph of \(\gamma_1 = \textbf{cdr}_{r_1}(\pi)\) has no unoriented components, and then choose an oriented vertex \(r_2\) of \(\gamma_1\) so that the overlap graph of \(\gamma_2 = \textbf{cdr}_{r_2}(\gamma_1)\) has no unoriented components. Then \(\gamma_2\) is \textbf{cdr} sortable and by Lemma 9.4 \((r_1, r_2, a_1)\) is a maximal oriented sequence of pointers for \(\pi\), whence \((a_1)\) is a maximal oriented sequence of pointers for \(\gamma_2\). Now \(\gamma_2\) has at least two adjacencies. Collapsing these adjacencies produces a signed permutation \(\gamma_2^*\) which has a maximal oriented sequence \((a_1^*)\) of pointers, and is a member of \(S^+_k\) for a \(k < n\). Applying \textbf{cdr}_{a_1^*} to \(\gamma_2^*\) produces a \textbf{cdr} fixed point \(\delta^*\) of \(\gamma^*\) which, by the induction hypothesis, is \textbf{cds} sortable. Reinstating the adjacencies we find that \(\delta\) is a \textbf{cdr} fixed point of \textbf{cdr}_{a_1}(\gamma_2),\) and is \textbf{cds} sortable. However, with \(\beta = \textbf{cdr}_{a_1}(\pi),\) we have by Lemma 9.2 that \(\delta = \textbf{cds}_{r_1,r_2}(\beta),\) and it follows that \(\beta\) is \textbf{cds} sortable.

**Case 2: \(m > 1\).** The induction hypothesis is that the statement is true for permutations \(\delta\) in \(S^+_k, k < n\), and all oriented sequences.

Subcase 2 (a): The overlap graph of \(\pi_1 = \textbf{cdr}_{a_1}(\pi)\) has no unoriented component. In this case, as \(\pi_1\) has an adjacency, we may reduce \(\pi_1\) through collapsing the adjacency, to an equivalent permutation \(\pi_1^*\) in \(S^+_{k-1}\) in which the oriented sequence corresponding to \((a_2, \ldots, a_m)\) sorts the equivalent permutation to the fixed point \(\beta^*\) corresponding to \(\beta\) under the same collapse of the same adjacency. By the induction hypothesis \(\beta^*\) is \textbf{cds}-sortable, and thus \(\beta\) is \textbf{cds}-sortable.
Finally explains the 2-to-1 ratio in cdr presentation can be computed by indiscriminate applications of cdr, for example Figures 9 and 10 of [14].

Theorem 9.6 (cdr Steps Theorem). Let π be a cdr sortable signed permutation. Suppose that k applications of cdr produces a cds fixed point β of π, and that m applications of cds to β results in the identity permutation. Then π is cdr sortable by k + 2m applications of cdr.

Thus, the number of applications of cdr required to sort a cdr sortable signed permutation can be computed by indiscriminate applications of cdr and cds. Theorem 9.6 also finally explains the 2-to-1 ratio in cdr and cds operations observed in [14] in constructing phylogenies among different Muller elements for several species of fruitflies, and justifies using this ratio in constructing distance matrices towards the phylogenetic analysis - see pp. 14-15, footnote 11 and for example Figures 9 and 10 of [14].
Theorem 10.1 (The cdr Parity Theorem). For each signed permutation $\pi$, the lengths of maximal sequences of pointers are of the same parity.

Proof. Let a signed permutation $\alpha \in S_n^\pm$ be given.

Case 1: $\alpha$ is cdr sortable: First note that any two total sequences of oriented vertices of the corresponding overlap graph are of the same length, by Theorem 4.3. Consider two sequences of pointers, say $S_1, S_2$, that each is maximal. Applications of cdr for these pointers thus result in cdr-fixed points $\beta_1$ and $\beta_2$ respectively. By Corollary 8.2 we can extend each of $S_1$ and $S_2$ individually by two pointers at a time until a total sequence of pointers is reached, meaning the identity permutation results from applications of cdr. By adding an even number of terms to a sequence, we do not change the parity of the lengths of these sequences. Thus, as the two sequences $S_1', S_2'$ which sort the permutation are the same length, and thus these lengths have same parity, also the lengths of $S_1$ and $S_2$ are the same parity. □

Case 2: $\alpha$ is not cdr-sortable. By Theorem 4.2 the overlap graph of $\alpha$ has unoriented components. The only pointers to which cdr can be applied are vertices in the oriented components of the overlap graph of $\alpha$. Consider some $\alpha'$ that has the same move graph as $\alpha$ without the unoriented component. Then $\alpha'$ has the same sequences of moves possible as $\alpha$. As $\alpha'$ does not have an unoriented component, Case 1 applies, whence lengths of all sequences for $\alpha'$ are of the same parity.

Example 8.3 provides an illustration of the cdr parity phenomenon: cdr fixed points are reached after 1, 3 or 5 applications of cdr. The cdr Parity Theorem can also be applied to a question about combinatorial games based on cdr, defined for signed permutations in [1].

In the oriented graph context these games are defined as follows: Let an oriented graph $(G, f)$ on a finite set of vertices be given. Player ONE starts the game by selecting an oriented vertex $o_1$ and computing the graph $(G_1, f_1) = gcdr((G, f), o_1)$. Then player TWO selects an oriented vertex $t_1$ of $(G_1, f_1)$ and computes the graph $(G_2, f_2) = gcdr((G_1, f_1), t_1)$, and so on. The game continues until no more legal moves are possible - i.e., there are no more oriented vertices left in the graph.

The normal play rule version of the game is denoted $N((G, f), gcdr)$. In this game, the player able to make the last legal move wins. In the special case when the graph $(G, f)$ is the oriented overlap graph of a signed permutation $\alpha$, this game is denoted $N(\alpha, cdr)$. In the misere play version, the player making the last legal move looses. The corresponding notation for the misere play games is $M((G, f), gcdr)$ or $M(\alpha, cdr)$.

Thus, the normal and misere version of the cdr game on signed permutations are solved: In order to determine who has the winning strategy, just determine the length of a single play of the game. This shows that decision problems D.6 and D.7 of [1] are linear time decision problems.

11. The ciliates

According to the model described in [16] the micronuclear precursors of genes in ciliates should be sortable or reverse sortable by cdr or cds to produce functional genes. For some indication of the scope of sorting required during ciliate micronuclear decryption, consider the following: Findings reported in [1] indicate that at least 3593 genes on 2818 chromosomes of the ciliate Sterkiella histriomuscorum have encrypted micronuclear precursors. 1676 of
these encrypted precursors contain at least one inverted element. [4] also reports an example of an encrypted precursor organized into 245 precursor segments.

Sample genes from ciliates show that both the \texttt{cdr} and the \texttt{cds} operations are necessary to sort or reverse sort signed permutations representing micronuclear precursors of genes. We discuss a small sample of micronuclear precursors, and their sortability by \texttt{cdr} and \texttt{cds}. More examples can be found at the ciliate genome rearrangement database [6].

\textbf{Example: The Actin I gene} \\
\textit{Oxytricha nova}: Decryption of the micronuclear precursor of the Actin I gene in the ciliate species \textit{Oxytricha nova}, reported in [11] (see Fig. 3 there), requires sorting of the signed permutation $\alpha = [3, 5, 4, 6, 8, -2, 1, 7]$. The overlap graph of $\alpha$, shown in Figure 11, has an unoriented component. $\alpha$ is neither \texttt{cdr} sortable, nor \texttt{cds}-sortable, but requires both \texttt{cdr} and \texttt{cds} for sorting. Applications of \texttt{cdr} terminate in the permutation $[-8, -7, -6, -4, -5, -3, -2, -1]$, which has item $-5$ out of order. An application of \texttt{cds} produces the permutation $[-8, -7, -6, -5, -4, -3, -2, -1]$, which represents a functional Actin I gene in the organism.

\textit{Uroleptus pisces}: The structures of the micronuclear precursors of the Actin I gene for \textit{U. pisces} 1 and \textit{U. pisces} 2, reported in [7], were given in the introduction. We treat $\alpha_1 = [1, 3, -7, -5, 14, 2, 4, 6, 9, 12, -11, -8, 13, 15, -10]$, the precursor for \textit{U. pisces} 1. To determine \texttt{cdr} sortability of $\alpha_1$ we construct the overlap graph of $\alpha_1$, shown in Figure 12.

The overlap graph of $\alpha_1$ is an oriented graph and its only component is oriented. By Theorem 4.2 $\alpha_1$ is \texttt{cdr} sortable. As the reader may verify, applying \texttt{cdr} in order for the following ordered length 14 sequence of pointers accomplishes such a sorting:

\[(4,5) (3,4) (2,3) (5,6) (6,7) (1,2) (7,8) (8,9) (9,10) (11,12) (12,13) (13,14) (14,15) (10,11)\]

By Theorem 4.3 any sequence of applications of \texttt{cdr} that sorts permutation $\alpha_1$ will use exactly fourteen pointers. But innocent-looking deviations from a sequence of pointers supporting a successful \texttt{cdr} sorting of $\alpha_1$ may result in a failed sorting. For example, if in the sequence of pointers above right after \texttt{cdr} has been applied to pointer (12,13) it is applied to pointer (10,11) (supporting a legitimate \texttt{cdr} application at that stage), the result will be the unsorted permutation

$$\beta = [1 2 3 4 5 6 7 8 9 1 0 1 1 2 1 3 1 5 1 4]$$
Figure 12. Overlap graph for α₁. Filled vertices are oriented.

How does the ciliate decryptome resolve this problem? The **cds Rescue Theorem**, Theorem 9.5 proves that the **cdr** fixed points encountered are in fact **cds** sortable. In this particular example, **cds**(α₁, β) is the identity permutation. Thus the **cds** sorting operation assures that decryption of α₁ does not fail. Moreover, this theorem and the **cds** Inevitability Theorem imply that indiscriminate applications of **cdr** and of **cds** will succeed in sorting α₁ (and any **cdr** sortable micronuclear gene pattern).

**Example: αTBP gene in *O. trifallax***

For some permutations that have been reported in extant ciliate species, **cds** is both necessary and sufficient: For example the micronuclear precursor of the α Telomere Binding Protein gene in *Oxytricha trifallax* is represented by the permutation [1, 3, 5, 7, 9, 11, 2, 4, 6, 8, 10, 12] has this feature. This permutation was reported in [17].

DNA polymerase α gene in *O. trifallax*

Likewise, for some signed permutations **cdr** is both necessary and sufficient to sort or reverse sort the corresponding permutation, and no application of **cds** would contribute to its sorting or reverse sorting. For this discussion, define for each n the signed permutations

\[ \sigma_n = [-2n, \ldots, -2, 1, 3, \ldots, 2n-1, 2n+1]. \]

and

\[ \tau_n = [-2n, \ldots, -2, 1, 3, \ldots, 2n-1]. \]

The oriented overlap graph of \( \sigma_n \) consists of 2n isolated oriented vertices, while the oriented overlap graph of \( \tau_n \) consists of 2n – 1 isolated oriented vertices. Observe that any pointer is eligible for a **cdr** move, and that this oriented graph has isolated vertices only. Thus, no applications of **cds** to any intermediaries contributes to the sorting process. The signed permutations \( \sigma_n \) and \( \tau_n \) also have the feature that for any selection of a pointer, a corresponding application of **cdr** can be performed. Each \( \sigma_n \) is **cdr** sortable, while each \( \tau_n \) is **cdr** reverse-sortable.
These permutations, intriguingly, occur in the micronuclear precursors of macro nuclear genes of certain ciliate species. For some species of *Uroleptus*, the DNA polymerase α gene is $\sigma_{16}$, while for *Paraurystola weissei* the same precursor is $\sigma_{20}$. In *Oxytricha nova* the micronuclear precursor is $\tau_{21}$. In *Oxytricha trifallax* the micronuclear precursor is $\sigma_{21}$. Accession numbers for these DNA polymerase α macronuclear and corresponding micronuclear precursors in [10] are

| Species                      | Macronuclear   | Micronuclear   |
|------------------------------|----------------|----------------|
| *Uroleptus*                  | AY293852       | AY293850       |
| *Paraurystola weissei*       | Y293806        | AY293805       |
| *Oxytricha trifallax*        | EF693893.1     | DQ525914.1     |

12. Concluding Remarks

Unless the applications of cdr operations during micronuclear decryption follow a strategy not yet discovered in the laboratory, it should be expected that cdr fixed points other than the identity are often encountered among intermediates of the decryption process. It would be interesting to determine whether cdr fixed points of cdr sortable micronuclear gene patterns actually do occur among intermediates of the decryption process.

Results reported in [4] (see Figure 3 A, B and C there) suggest that the model of [15, 16] based on just the cdr and cds sorting operations might require an additional operation to successfully sort newly observed encryption patterns. The permutation representing the red micronuclear precursor in Figure 3 B, and the permutation representing the gold micronuclear precursor in Figure 3 C are cds fixed points. It would be interesting to learn how these two particular micronuclear precursors are in fact processed by the ciliate decryptome. In [8] and some earlier papers it is assumed that besides cdr and cds, there is an additional sorting operation, boundary ld, that would sort a cds fixed point to the identity permutation. To our knowledge it has not been experimentally confirmed that this operation occurs during ciliate micronuclear decryption, nor has a satisfactory molecular mechanism for this operation in combination with cdr and cds been described.

13. Acknowledgements

The research represented in this paper was funded by an NSF REU grant DMS 1359425, by Boise State University and by the Department of Mathematics at Boise State University.

References

[1] K. Adamyk, E. Holmes, G. Mayfield, D.J. Moritz, M. Scheepers, B.E. Tenner and H. Wauck, *Sorting permutations: Games, genomes and cycles*, arXiv1410.2353
[2] A. Bergeron, *A very elementary presentation of the Hannenhalli-Pevzner theory*, *Discrete Applied Mathematics* 146:2 (2005), 134 - 145
[3] A. Bergeron, S. Heber and J. Stoye, *Common intervals and sorting by reversals: a marriage of necessity*, *Bioinformatics* 18:2 (2002), S54 - S63.
[4] X. Chen, J.R. Bracht, A.D. Goldman, E. Dolzhenko, D.M. Clay, E.C. Swart, D.H. Perlman, T.G. Doak, A. Stuart, C.T. Amemiya, R.P. Sebra, and L.F. Landweber, *The Architecture of a Scrambled Genome Reveals Massive Levels of Genomic Rearrangement during Development*, *Cell* 158 (2014), 1187–1198
[5] D.A. Christie, *Sorting permutations by block-interchanges*, *Information Processing Letters* 60 (1996), 165 - 169.
[6] Ciliate Genome Rearrangement Database, [http://oxytricha.princeton.edu/mds jes_db/index.html](http://oxytricha.princeton.edu/mds jes_db/index.html)
[7] A.B. Dalby and D.M. Prescott, *The scrambled Actin I gene in Uroleptus pisces*, Chromosoma 112 (2004), 247 - 254.
[8] A. Ehrenfeucht, T. Harju, I. Petre, D. M. Prescott, and G. Rozenberg, *Computation in Living Cells: Gene Assembly in Ciliates*, Springer: Natural Computing Series, 2003.
[9] G. Fertin, A. Labarre, I. Rusu, E. Tannier and S. Vialette, *Combinatorics of Genome Rearrangements*, The MIT Press, 2009.
[10] GenBank, a database of the National Center for Biotechnology Information, maintained electronically at http://www.ncbi.nlm.nih.gov/genbank/
[11] A.F. Greslin, D.M. Prescott, Y. Oka, S.H. Loukin and J.C. Chappell, *Reordering of nine exons is necessary to for a functional Actin gene in Oxytricha nova*, Proceedings of the National Academies of Science of the United States of America 86:16 (1989), 6264 - 6268.
[12] S. Hannenhalli and P.A. Pevzner, *Transforming cabbage into turnip: Polynomial algorithm for sorting signed permutations by reversals*, Journal of the ACM 46:1 (1999), 1 - 27.
[13] I. Harju, T. Petre and G. Rozenberg, *Two models for gene assembly in ciliates*, Turku Centre for Computer Science Technical Reports 406, 2004
[14] J. Herlin, A. Nelson and M. Scheepers, *Using Ciliate Operations to construct Chromosome Phylogenies*, Involve 9:1 (2016), 1 - 26.
[15] D.M. Prescott, A. Ehrenfeucht and G. Rozenberg, *Molecular operations for DNA processing in hypotrichous ciliates*, European Journal of Protistology 37 (2001), 241 - 260.
[16] D.M. Prescott, A. Ehrenfeucht and G. Rozenberg, *Template-guided recombination for IES elimination and unscrambling of genes in stichotrichous ciliates*, Journal of Theoretical Biology 222 (2003), 323 - 330.
[17] J.D. Prescott, M.L. DuBois and D.M. Prescott, *Evolution of the scrambled germline germline gene encoding α-telomere binding protein in three hypotrichous ciliates*, Chromosoma 107 (1998), 293 - 303.
[18] A.H. Sturtevant and Th. Dobzhansky, Inversions in the third chromosome of wild races of Drosophila pseudoobscura, and their use in the study of the history of the species, Proceedings of the National Academy of Sciences 22 (1936) 448 – 450.
[19] E. Tannier, A. Bergeron and M-F. Sagot, *Advances on sorting by reversals*, Discrete Applied Mathematics 155 (2007), 881 - 888.