Rearrangement Models and Single-Cut Operations

Paul Medvedev[1] and Jens Stoye[2]

[1]Department of Computer Science, University of Toronto,
pashadag@cs.toronto.edu
[2]Technische Fakultät, Universität Bielefeld, Germany,
stoye@techfak.uni-bielefeld.de

Abstract. There have been many widely used genome rearrangement models, such as reversals, Hannenhalli-Pevzner, and double-cut and join. Though each one can be precisely defined, the general notion of a model remains undefined. In this paper, we give a formal set-theoretic definition, which allows us to investigate and prove relationships between distances under various existing and new models. We also initiate the formal study of single-cut operations by giving a linear time algorithm for the distance problem under a new single-cut and join model.

1 Introduction

In 1938, Dobzhansky and Sturtevant first noticed that the pattern of large scale rare events, called genome rearrangements, can serve as an indicator of the evolutionary distance between two species [9]. With the pioneering work of Sankoff and colleagues to formulate the question of evolutionary distance in purely combinatorial terms [23, 22], the mathematical study of genome rearrangements was initiated. Here, the evolutionary distance is determined as the smallest number of rearrangements needed to transform one genome (abstracted as a gene-order) into another. This has given rise to numerous combinatorial problems, including distance, median, aliquoting, and halving problems, which are used to build phylogenetic trees and infer other kinds of evolutionary properties.

An underlying challenge of such approaches is to define an appropriate model, which specifies the kinds of rearrangements allowed. On one hand, the model should be as accurate as possible, including all the possible underlying biological events and weights which reflect their likelihood. On the other, answering questions like the median or distance can be computationally intractable for many models. The trade-offs between these, as well as other, considerations decide which rearrangement model is best suited for the desired type of analysis.

Though the ideas of genomes and rearrangements are inherently biological, they require precise mathematical definitions for the purposes of combinatorial analysis. Earlier definitions of genomes as signed permutations did not generalize well to genomes with duplicates, but recently a more general set-theoretic definition in terms of adjacencies has become used [5]. However, though particular models, like HP or DCJ, have their own precise definitions, the notion of a model, in general, remains undefined. In this paper, we give such a definition and show how current rearrangement models can be defined within our framework. This allows us to investigate and prove things about the relationship between sorting distances under different models, which we present in combination with what is already known to give an exposition of current results.

Recently, it was observed that most of the events in a parsimonious evolution scenario between human and mouse were operations which cut the genome in only one place, such as fusions, fissions, semi-translocations, and affix reversals (reversals which include a telomere) [7]. Such scenarios have applicability to the breakpoint reuse debate [2, 21, 24, 7] since they can suggest a low rate of reuse. In this paper, we initiate the formal study of such single-cut operations by giving a linear time algorithm to find the minimum distance under a new single-cut and join (SCJ) model[1] and using it to determine the SCJ distance between the human and several other organisms.

[1] Note that here SCJ refers to single-cut **and** join, as opposed to single-cut **or** join which was recently introduced by Feijão and Meidanis [10].

2 Preliminaries

We begin by giving the standard definition of a genome, consistent with [5]. We represent the genes by a finite subset of the natural numbers, $N \subset \mathbb{N}$. For a gene $g \in N$, there is a corresponding *head* g_{h} and *tail* g_{t}, which are together referred to as the *extremities* of g. The set of all extremities of all genes in N is called N_{ext}. The set $\{p, q\}$, where p and q are extremities, is called an *adjacency*. We denote by N_{adj} the set of all possible adjacencies of N. The one-element set $\{p\}$, where p is an extremity, is called a *telomere*. We denote by N_{tel} the set of all possible telomeres of N. Telomeres and adjacencies are collectively referred to as *points*. A *genome* $G \subseteq N_{\text{adj}} \cup N_{\text{tel}}$ is a set of points such that each extremity of a gene appears exactly once[2]:

$$\bigcup_{x \in G} x = N_{\text{ext}} \text{ and for all } x, y \in G, x \cap y = \emptyset$$

For brevity, we will sometimes use signed permutation notation to describe a uni-chromosomal linear genome, such as $G = (1, -2, -3, 4)$; however, this is just a notation and the underlying representation of the genome is always as a set of points. We denote by $N(G)$ the set of genes underlying the genome G. Finally, we define $\mathcal{G}$ to be the set of all possible genomes over all possible gene sets ($\mathcal{G}$ is a countable infinite set).

Though this definition of a genome does not immediately reflect the notion of chromosomes or gene-orders, these are reflected as properties of the genome graph. Given a genome G, its *genome graph* is an undirected graph whose vertices are exactly the points of G. The edges are exactly the genes of G, where edge g connects the two vertices that contain the extremities of g (this may be a loop). It is easy to show that the genome graph is a collection of cycles and paths [5].

We can now define a *chromosome* as a connected component in the genome graph. We also say that a sequence of extremities $p_1, \ldots, p_m$ is *ordered* if there exists a path which traverses the vertices associated with the extremities in the given order. Note that questions like the number of chromosomes, whether two genes lie on the same chromosome, or whether extremities are ordered can all be answered in linear time by constructing and analyzing the genome graph.

Another useful graph is the *adjacency graph* [5]. For two genomes A and B, $AG(A, B)$ is an undirected, bipartite multi-graph whose vertices are the points of A and B. For each $x \in A$ and $y \in B$ there are $|x \cap y|$ edges between x and y. It is not difficult to show that this graph is a vertex-disjoint collection of paths and even-cycles and can be constructed in linear time [5]. We denote by $C_s(A, B)$, $C_l(A, B)$, and $I(A, B)$ the number of short cycles (length two), long cycles (length greater than 2), and odd paths in $AG(A, B)$, respectively. We use the term A-path to refer to a path that has at least one endpoint in A, and BB-path to one with both endpoints in B.

3 Models, operations, and events

We now present a formal treatment of rearrangement models, beginning with the main definition:

Definition 1 (Rearrangement Models, Operations and Events). *A rearrangement operation (also called a model) is a binary relation $\mathcal{R} \subseteq \mathcal{G} \times \mathcal{G}$. A rearrangement event is a pair $R = (G_1, G_2) \in \mathcal{G} \times \mathcal{G}$.[3] We say that R is an $\mathcal{R}$ event if $R \in \mathcal{R}$.*

For example, if we have genes $\{a, b, c\}$ and a genome $G = \{\{a_h\}, \{a_t, b_h\}, \{b_t\}, \{c_h\}, \{c_t\}\}$, then a possible event is $R = (G, \{\{a_h\}, \{a_t, b_h\}, \{b_t, c_h\}, \{c_t\}\})$. This event has the effect of fusing the two chromosomes of G. On the other hand, a fusion operation $\mathcal{R}$ is given by the set of all pairs (G_1, G_2) such that there exist extremities $x, y \in N(G_1)_{\text{ext}}$ and $G_2 \cup \{\{x\}, \{y\}\} = G_1 \cup \{\{x, y\}\}$. It is easy to see that

[2] Though this paper focuses on genomes without duplicate genes, this definition could be extended to the more general case by treating the set of genes and its corresponding derivatives as multisets, including the genome.

[3] Alternatively, we can treat R as a function in the standard way of viewing relations as functions. Namely, $R : \mathcal{G} \cup \{\emptyset\} \longrightarrow \mathcal{G} \cup \{\emptyset\}$ where $R(C) = G_2$ if $C = G_1$ and $R(C) = \emptyset$ otherwise.

R is an $\mathcal{R}$-event. Thus the operation $\mathcal{R}$ captures the general notion of a fusion as a type of rearrangement, while the event R captures this particular instance of a fusion. Most current literature does not make a formal distinction between types of rearrangements (which we call operations) and their particular instances (which we call events), but this is necessary for defining the notion of a rearrangement model.

Current literature often makes the informal distinction between models, such as DCJ or HP, and operations, such as reversals or fusions. Operations are considered more biologically atomic, with a model being a combination of these atomic operations. Here, we maintain this notational consistency; however, we note that the terms model and operation are mathematically equivalent.

In this paper, we focus on the double-cut and join (DCJ) model and its subsets (referred to as *submodels* in the context of models).

Definition 2 (DCJ). *Let G_1 and G_2 be two genomes with equal gene content ($N(G_1) = N(G_2)$). Then $(G_1, G_2) \in$ DCJ if and only if there exist extremities p, q, r, s such that one of the following holds:*

(a) $G_2 \cup \{\{p, q\}, \{r, s\}\} = G_1 \cup \{\{p, r\}, \{q, s\}\}$
(b) $G_2 \cup \{\{p, q\}, \{r\}\} = G_1 \cup \{\{p, r\}, \{q\}\}$
(c) $G_2 \cup \{\{q\}, \{r\}\} = G_1 \cup \{\{q, r\}\}$
(d) $G_2 \cup \{\{q, r\}\} = G_1 \cup \{\{q\}, \{r\}\}$

This definition is equivalent to the one given in [27, 5]. A more intuitive interpretation of, for example, (a), is that the event replaces the adjacencies $\{p, q\}$ and $\{r, s\}$ in G_1 with $\{p, r\}$ and $\{q, s\}$ in G_2.

Note that an event that satisfies one of the conditions (b)-(d) of the DCJ model only cuts the genome in one place. These events define the submodel of DCJ called single-cut and join (SCJ), which we will study in Section 6. Other operations, such as reversals, can be defined in a similar manner, though we do not do it here for conciseness. We will, however, define how to restrict a model so that it only deals with linear and/or uni-chromosomal genomes:

Definition 3. *Given a model M, let*

- $M_{lin} = \{(G_1, G_2) \mid (G_1, G_2) \in M$ *and G_1 and G_2 are linear*$\}$.
- $M_{uni} = \{(G_1, G_2) \mid (G_1, G_2) \in M$ *and G_1 and G_2 are uni-chromosomal*$\}$.

There are many questions one can pose within any model, including sorting, distance, median, halving, or aliquoting. We will focus on the sorting and distance problems here:

Definition 4 (Sorting Sequence and Distance). *A sequence of events $R_1, R_2, \ldots, R_m$ sorts G_1 into G_2 if $G_2 = R_m(\ldots(R_1(G_1)))$. The sorting distance between G_1 and G_2 under a model $\mathcal{R}$, denoted by $d_{\mathcal{R}}(G_1, G_2)$, is the length of the shortest sorting sequence such that all R_i are $\mathcal{R}$ events. If such a sequence does not exist then we say $d_{\mathcal{R}}(G_1, G_2) = \infty$.*

Given two genomes, we can either find a shortest sorting sequence or just the sorting distance. Since the number of possible events in each step is polynomial, if we have a polynomial time algorithm for distance, then we also have one for sorting [13]. That is why when discussing poly-time complexity, we will focus only on the distance problem. Note however that the precise complexities may differ, as for example is the case for the reversal model, where the distance can be computed in linear time [4] while the best-known sorting algorithms have worst-case time complexity $O(n^{3/2}\sqrt{\log n})$ [25].

4 Submodels of double-cut and join

Motivated by different biological systems, several distinct rearrangement models have been studied. These models differ in various aspects, spanning a whole space of genome rearrangement models. The three most relevant dimensions of this space are (i) the number of chromosomes a genome may have, (ii) the shape that the chromosomes may have (i.e. linear or circular), and (iii) the maximum number of chromosome cuts (and

	Model					
Operation	DCJ	HP	SCJ	SCJ_{lin}	$(\text{SCJ}_{\text{uni}})_{\text{lin}}$	
PROPER TRANSLOCATION	●	●				
SEMI TRANSLOCATION	●	●	●	●		
PATH FISSION/FUSION	●	●	●	●		
EXCISION/INTEGRATION	●					
REVERSAL	●	●	○	○		○
CIRCULARIZATION/LINEARIZATION	●		●			
CYCLE FISSION/FUSION	●		○			

Table 1. A description of some of the models in terms of the elementary operations defined in [5]. A dark bullet means that the operation is fully contained in the model, no bullet means that the operation is disjoint with the model, and an empty bullet means that some but not all of the operation is contained in the model. Furthermore, each model is precisely the union of the operations specified by the bullets.

joins) an operation may perform[4]. This three-dimensional space is visualized in Fig. 1. Each of the corners of the cube can be formally defined by deriving a submodel from either DCJ or SCJ using the linear and/or uni-chromosomal restrictions. For example, $\text{HP} = \text{DCJ}_{\text{lin}}$, and the front bottom left corner is $(\text{SCJ}_{\text{uni}})_{\text{lin}}$ One can also think of these models in terms of the operations they allow, which is shown in Table 1.

Of the corners of the cube visualized in Fig. 1, some are of particular interest and thus have been studied more than others. The first model to be studied was the REV model, where the only allowed operation was the reversal. This model can only be used to sort linear uni-chromosomal genomes, since a reversal can never change the number of chromosomes or make them circular. The biological motivation for this model goes back to Nadeau and Taylor [19] and it was first formally modeled by Sankoff [22]. The first polynomial-time algorithm for computing the reversal distance and solving the reversal model was given by Hannenhalli and Pevzner [13] in 1995.

In the same year, Hannenhalli and Pevzner also looked at a model where multiple linear chromosomes are allowed, as is often the case in eukaryotic genomes [12]. After its authors, the resulting model is called the HP model. A combination of work showed that the distance under HP can be computed in poly-time [12, 26, 20, 14, 6].

A more recently introduced model is the double-cut and join (DCJ), which encompasses all events that can be achieved by first cutting the genome in up to two places, and then rejoining them in different combinations [27]. Though such a model is less biologically realistic than the HP model, there are fast algorithms for solving it [5] which have made it useful as an efficient approximation for the HP distance [1, 15, 18]. The DCJ model is a superset of all the other models in the cube, and is thus the most general.

The REV, HP, and DCJ models are all double-cut models, in that they allow for the cutting of the genome in two places. However, one can also consider models where only one cut is allowed. These make up the bottom plane of the cube, with the most general of these being the already defined SCJ model. These models have not yet been studied, since they are quite restrictive. However, we became aware of their relevance when we looked at certain rearrangement scenarios in eukaryotic evolution. In particular, while studying rearrangement scenarios between human and mouse with a minimum number of breakpoint reuses [7] it was observed that most of the events (213 out of 246) were single-cut (186 semi-translocations and affix reversals, 15 fissions and 12 fusions). This observation raised our interest in studying SCJ and its submodels.

In Section 6, we give a linear time algorithm for the sorting distance in the SCJ model. When restricted to linear chromosomes (SCJ_{lin}), we have the single-cut equivalent of the HP model, allowing fissions, fusions, semi-translocations, and affix reversals. The complexity of this model is unknown. The even more restrictive $(\text{SCJ}_{\text{uni}})_{\text{lin}}$ model consists of only affix reversals, which reverse a prefix/suffix of a chromosome, and is the single-cut equivalent of the REV model. There is a simple 2-approximation algorithm for the sorting distance,

[4] In this paper, we focus on double and single-cut operations. However, more general k-cut operations have also been considered [3].

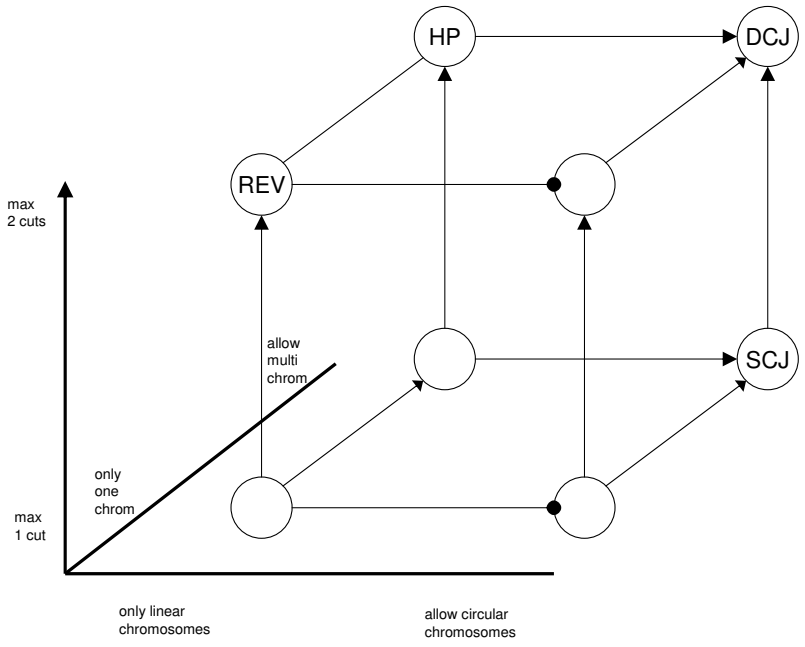


Fig. 1. The space of DCJ genome rearrangement submodels. An arrow on an edge from M to M' indicates that there exists a pair of genomes that are sortable under both models and whose distance under M is strictly more than under M'. An edge between M and M' with a circular ending at M' indicates that if a pair of genomes is sortable under M, then its distances under M and M' are the same.

which increases the number of short cycles by at least one every two steps. However, the complexity of the problem remains open. It is related to the problem of sorting burnt pancakes [11, 8], which is similar except that the chromosome has an orientation and only prefix reversals are allowed. The complexity of this problem is also open.

5 Sorting distance relationships

In this section, we study the relationship between sorting distances under the various models represented in the cube of Figure 1. We start with an easy observation, which we sometimes may use without explicitly stating it:

Lemma 1 (Submodel Lemma). *For any two models M and M' with $M' \subseteq M$, and all genomes G_1, G_2,*

$$d_M(G_1, G_2) \leq d_{M'}(G_1, G_2).$$

Proof. Any sorting sequence under M' is by definition also a sorting sequence under M. □

Corollary 1. *For all genomes G_1, G_2 and models M,*

- $d_{\text{DCJ}}(G_1, G_2) \leq d_{\text{SCJ}}(G_1, G_2)$

– $d_M(G_1, G_2) \leq d_{M_{lin}}(G_1, G_2)$
– $d_M(G_1, G_2) \leq d_{M_{uni}}(G_1, G_2)$

In fact, if you compare two models that are connected by an edge of the cube, then the one that is furthest from the origin is at least as powerful as the closer one. An interesting question is when the two models are equally powerful – that is, if a pair of genomes is sortable under both models, then the distances are the same. We first study if the restriction to linear or uni-chromosomal genomes makes the DCJ or SCJ models less powerful. Of course, it is clear from the definitions that, for example, a multi-chromosomal genome can be sorted by SCJ and cannot be sorted by SCJ_{uni}. However, we are interested if there are genomes which can be sorted by both models, but with fewer steps in one than the other.

Lemma 2. *There exist two genomes G_1 and G_2 such that*

– $d_{\text{SCJ}}(G_1, G_2) < d_{\text{SCJ}_{lin}}(G_1, G_2) < \infty$
– $d_{\text{SCJ}}(G_1, G_2) < d_{\text{SCJ}_{uni}}(G_1, G_2) < \infty$
– $d_{\text{DCJ}}(G_1, G_2) < d_{\text{DCJ}_{uni}}(G_1, G_2) < \infty$

Proof. Let $G_1 = (1, 3, 2)$ and $G_2 = (1, 2, 3)$. We can sort G_1 into G_2 using two SCJ events. First, we make an excision by replacing points $\{1_h, 3_t\}$ and $\{2_h\}$ with $\{1_h\}$ and $\{2_h, 3_t\}$. Second, we make an integration by replacing points $\{1_h\}$ and $\{3_h, 2_t\}$ with $\{1_h, 2_t\}$ and $\{3_h\}$. However, there does not exist a sorting sequence of length two under either SCJ_{lin}, SCJ_{uni}, or DCJ_{uni} models, though the genomes are clearly sortable under all these models. □

To complete the picture, it is already known that there are genomes that are sortable in HP but require more steps than in DCJ (see [6] for an example).

We next look if the flexibility of double-cut operations makes the models in the top plane more powerful than their respective counterparts in the bottom plane. There is a simple example that answers this question in the affirmative.

Lemma 3. *There exist two genomes G_1 and G_2 such that*

– $d_{\text{REV}}(G_1, G_2) < d_{(\text{SCJ}_{uni})_{lin}}(G_1, G_2) < \infty$
– $d_{\text{HP}}(G_1, G_2) < d_{\text{SCJ}_{lin}}(G_1, G_2) < \infty$
– $d_{\text{DCJ}_{uni}}(G_1, G_2) < d_{\text{SCJ}_{uni}}(G_1, G_2) < \infty$
– $d_{\text{DCJ}}(G_1, G_2) < d_{\text{SCJ}}(G_1, G_2) < \infty$

Proof. Let $G_1 = (1, -2, 3)$ and $G_2 = (1, 2, 3)$. We can sort G_1 into G_2 using just one event in the REV model, while there is no single SCJ event that does this. However, G_1 is sortable into G_2 using affix reversals. The lemma follows by applying the Submodel Lemma. □

We now compare $(\text{SCJ}_{\text{uni}})_{\text{lin}}$ with SCJ_{lin} to determine if the flexibility to create additional chromosomes in intermediate steps adds power when we are restricted to single-cut operations and linear genomes.

Lemma 4. *There exist two genomes G_1 and G_2 such that*

$$d_{\text{SCJ}_{lin}}(G_1, G_2) < d_{(\text{SCJ}_{uni})_{lin}}(G_1, G_2) < \infty.$$

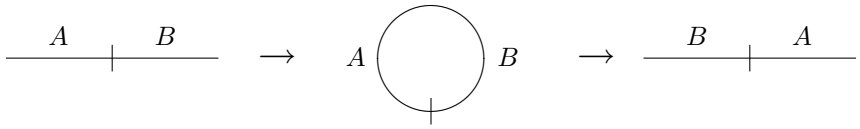
Proof. Let $G_1 = (1, -2, -3, 4)$ and $G_2 = (1, 2, 3, 4)$. There exists a sorting sequence of length 4 under SCJ_{lin} that makes a fission between -2 and -3, two affix reversals of genes 2 and 3, respectively, and a final fusion. However, one can check that using only affix reversals the sorting distance is 6. □

In some cases, however, additional flexibility does not add power to the model. Consider the SCJ_{uni} model, which differs from the $(\text{SCJ}_{\text{uni}})_{\text{lin}}$ model in that, besides affix reversals, it allows the circularization and linearization of the chromosome. This obviously allows sorting a linear chromosome into a circular one, something that REV does not allow. However, for genomes that are sortable under both models, we can show that circularization cannot help to decrease the distance.

Lemma 5. *For two uni-chromosomal linear genomes G_1 and G_2, we have*

$$d_{\text{SCJ}_{uni}}(G_1, G_2) = d_{(\text{SCJ}_{uni})_{lin}}(G_1, G_2).$$

Proof. We show that for any optimal SCJ_{uni} sorting scenario that creates a circular chromosome in an intermediate step, there exists a sorting scenario of equal length without a circular chromosome. Since the only SCJ operation that can be performed on a uni-chromosomal circular genome is to linearize it, we know that every circularization is immediately followed by a linearization. Thus, w.l.o.g. the situation can be described as an exchange of a prefix A and a suffix B:



However, the same effect can be achieved by two affix reversals, namely first reversing A and then B:

□

A similar situation occurs when we add circularization and linearization to the reversal model, though the proof is more involved:

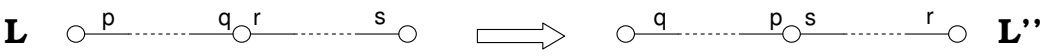
Lemma 6. *For two uni-chromosomal linear genomes G_1 and G_2, we have*

$$d_{\text{DCJ}_{uni}}(G_1, G_2) = d_{\text{REV}}(G_1, G_2).$$

Proof. Consider an optimal DCJ_{uni} sorting sequence S that has the smallest possible number of circularizations. We show, by contradiction, that this number is zero, proving the lemma. Let L be the genome prior to the first circularization, let C be the one right after it, let C' be the genome right before the first linearization, and let L' be the one right after it. Let d be the length of the sorting sequence between C and C' (these must be reversals).

We will apply Theorem 3.2 from [17], which states that the reversal distance between any two circular chromosomes (C and C' in our case) is the same as the reversal distance between their linearizations, if they share a telomere. If L and L' share a telomere, then there is a sorting sequence with shorter length that replaces $d + 2$ events between them with d reversals, which contradicts the optimality of S.

Suppose w.l.o.g that the two telomeres of L are $\{p\}$ and $\{s\}$, that $\{q\}$ is a telomere in L', that $\{q, r\}$ is an adjacency in L, and p, q, r, s are ordered in L. We can perform two reversals on L, the first one replacing $\{p\}$ and $\{q, r\}$ with $\{q\}$ and $\{p, r\}$, and the second replacing $\{p, r\}$ and $\{s\}$ with $\{p, s\}$ and $\{r\}$. The effect on the genome graph can be visualized as follows:



Note that if you circularize the resulting genome, L'', you get C, and L'' shares a telomere with L'. Therefore, by the theorem, there exist d reversals that sort L'' into L'. We can then get a new sorting sequence that replaces the $d+2$ events between L and L' with the two reversals described above followed by d reversals given by the theorem. This sorting sequence has the same length as S but has one less circularization, a contradiction. □

The results of this section are compactly summarized in Figure 1 by marking the endpoints on the edges of the cube. The Submodel Lemma implies other results which we have not explicitly stated but that can be

deduced by looking at the endpoints of the edges. For instance, there are genomes that are sortable under SCJ_{uni} that require less steps under SCJ_{lin}. Additionally some models which are not subsets of each other (like SCJ and HP) are incomparable. That is, there are genomes that are sortable under SCJ but require less steps under HP, and there are other genomes for which the opposite holds. The examples of $(1, 3, 2)$ and $(1, -2, 3)$ are enough to prove incomparability of SCJ with REV and with HP.

There is one edge of the cube with no marked endpoints, which represents the question of whether HP is more powerful to sort uni-chromosomal genomes than REV. Note that this is not trivial, because under HP we can split chromosomes in intermediate steps, which proved to give SCJ_{lin} more power than $(\text{SCJ}_{\text{uni}})_{\text{lin}}$ (Lemma 4). To the best of our knowledge, this question remains open.

6 Single-cut and join

We have already motivated the study of the sorting distance under the SCJ model, and in this section we give a linear time algorithm to compute it. Let A and B be two arbitrary genomes with the same underlying set of genes, N. We will use the following potential function in our analysis

$$\Phi(A, B) = |N| - I(A, B)/2 - C_s(A, B) + C_l(A, B)$$

First, we show that the potential function is 0 if and only if the two genomes are the same:

Lemma 7. *$A = B$ if and only if $\Phi(A, B) = 0$.*

Proof. The only if direction follows trivially from the definition of the adjacency graph. The if direction follows from a simple counting argument. Let a be the number of adjacencies in A, and t be the number of telomeres. By definition, $a + t/2 = |N|$. Since each short cycle accounts for one adjacency of A, and each odd path accounts for two telomeres of A, we have that $C_s(A, B) + I(A, B)/2 \leq a + t/2 = |N|$. Therefore, for the equality of the lemma to hold, we must have $C_l(A, B) = 0$, $C_s(A, B) = a$, and $I(A, B) = t$. This further implies, since each path is responsible for at least one telomere, and there are the same number of telomeres as paths, that all the paths must have length 1. Since $AG(A, B)$ contains only paths of length 1 and short cycles, we conclude that the points of A must be the same as the points of B. □

We can show using simple case analysis that Φ can decrease by at most one after any single event (proof omitted).

Lemma 8. *For all* SCJ *events R, $\Phi(R(A), B) - \Phi(A, B) \geq -1$.*

Combined with the fact that Φ cannot be negative, this gives a lower bound of $\Phi(A, B)$ on the sorting distance. We now consider Algorithm 1, whose cases are also illustrated in Figure 2.

Lemma 9. *Algorithm 1 terminates and outputs an* SCJ *sorting scenario of length $\Phi(A, B)$.*

Proof. First, we observe that one of the cases always applies since if $A \neq B$ then there must be at least one path of length greater than one or a long cycle. One can also verify that in each case, (A, A') is an SCJ event. Finally, we show that $\Phi(A', B) - \Phi(A, B) = -1$:

Case 1: A short cycle is created and the length of P decreases by two.
Case 2: An even path (P) is removed and a short cycle is created.
Case 3: An even path (P) is replaced by two odd paths.
Case 4: A long cycle is replaced by an even path.

□

Thus, we have

Theorem 1. $d_{\text{SCJ}}(A, B) = \Phi(A, B) = |N| - I(A, B)/2 - C_s(A, B) + C_l(A, B)$.

Algorithm 1 Algorithm for sorting under SCJ

while $A \neq B$ **do**
  if there exists an A-path P with length > 3 **then**
    Let p, q, r be the first three edges (from an arbitrary A end P).
    Let $A' = A \setminus \{\{p\}, \{q, r\}\} \cup \{\{r\}, \{p, q\}\}$.
  else if there exists an A-path P with length of 2 **then**
    Let p and q be its two edges.
    Let $A' = A \setminus \{\{p\}, \{q\}\} \cup \{\{p, q\}\}$.
  else if there exists a BB-path P **then**
    Let p and q be the first two edges (from an arbitrary end of P).
    Let $A' = A \setminus \{\{p, q\}\} \cup \{\{p\}, \{q\}\}$.
  else if there exists a long cycle **then**
    Let $\{p, q\}$ be a vertex of the cycle in A.
    Let $A' = A \setminus \{\{p, q\}\} \cup \{\{p\}, \{q\}\}$.
  end if
  Print A'.
  Let $A = A'$.
end while

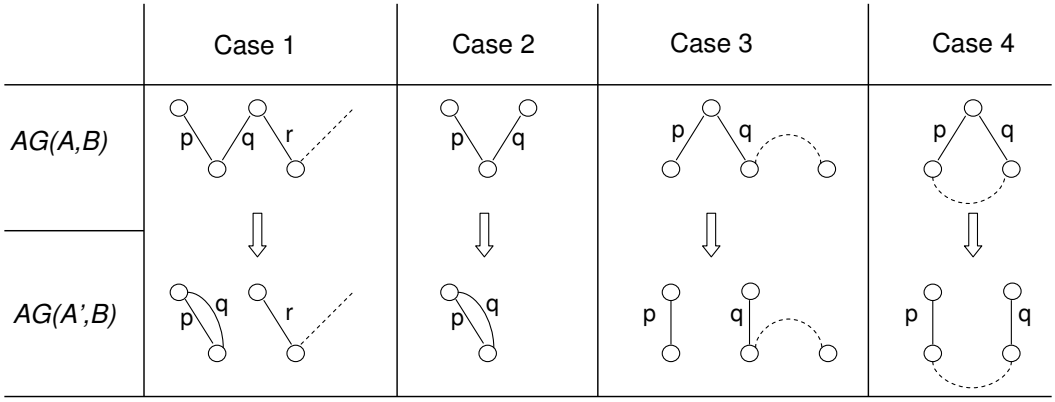


Fig. 2. The four cases of Algorithm 1.

Corollary 2. *$d_{\text{SCJ}}(A, B)$ is computable in $O(|N|)$ time.*

Note the similarity to the formula for the DCJ distance [5]:

$$d_{\text{DCJ}}(A, B) = |N| - I(A, B)/2 - C_s(A, B) - C_l(A, B)$$

Thus the difference between the SCJ and DCJ distances is $2C_l(A, B)$.

7 Experimental results

We performed six different comparisons, all with respect to the human. In the first, we took the 281 synteny blocks of the mouse-human comparison done in [21]. In the other five, we used the 1359 synteny blocks of the chimp, rhesus monkey, mouse, rat, and dog used in [16]. For each comparison, we computed the SCJ, DCJ, and HP distances. The HP distance was computed using GRIMM [26]. The results are shown in Table 2.

One immediately sees that the SCJ scenarios are far from parsimonious relative to the HP model. However, we stress that the goal of the SCJ model is to explore the importance of double-cut operations in evolution,

		Model			Ratio
Organism	#Blocks	DCJ	HP	SCJ	SCJ/ HP
Mouse [21]	281	246	246	300	1.2
Chimp	1359	22	23	58	2.5
Rhesus Monkey	1359	106	110	224	2.0
Mouse [16]	1359	408	409	642	1.6
Rat	1359	707	753	1291	1.7
Dog	1359	291	295	523	1.8

Table 2. Rearrangement distances under different models from different organisms to the human.

and not to be a realistic evolutionary model. It can be an indicator of how many double-cut operations are really an advantage and how many are just an alternative that can be avoided. For example, consider that the difference between the HP and SCJ distances is 150% in the chimp-human comparison, and 60% in the mouse[16]-human comparison. This might suggest that somehow single-cut operations play a lesser part in the chimp-human evolution than in the mouse-human evolution. We also notice that the ratio of the SCJ to HP distance in the mouse-human comparison is much lower (1.2 vs. 1.6) using the synteny blocks of [21] than using the synteny blocks of [16]. This suggests the sensitivity of this kind of breakpoint reuse analysis to synteny block partition.

In Section 5, we showed that there exist genomes for which the SCJ distance is smaller than the HP distance (for example $1, 3, 2$). However, in all the experimental results, the HP distance is always much smaller. This suggests that the SCJ operations not allowed by HP, such as excissions, integrations, circularizations, and linearizations, are infrequent relative to fissions, fusions, translocations, and reversals.

8 Conclusion

In this paper, we gave a formal set-theoretic definition of rearrangement models and operations, and used it to compare the power of various submodels of DCJ with uni-chromosomal and/or linear restrictions. We hope that the formal foundation for the notion of models will eventually lead to further insights into their relationships.

We also initiated the formal study of single-cut operations by giving a linear time algorithm for computing the distance under a new single-cut and join model. Many interesting algorithmic questions remain open, including the complexity of sorting using linear SCJ operations, and sorting using affix reversals.

Acknowledgements We would like to thank Anne Bergeron, and PM would like to also thank Allan Borodin and Michael Brudno, for useful discussions. Most of the work was done while PM was a member of the International NRW Graduate School in Bioinformatics and Genome Research, and the AG Genominformatik group at Bielefeld, while being additionally funded by a German Academic Exchange Service (DAAD) Research Grant. PM gratefully acknowledges the support of these organizations.

References

1. Z. Adam and D. Sankoff. The ABCs of MGR with DCJ. *Evol. Bioinform.*, 4:69–74, 2008.
2. M. A. Alekseyev and P. A. Pevzner. Are there rearrangement hotspots in the human genome? *PLoS Comput. Biol*, 3(11):e209, 11 2007.
3. M. A. Alekseyev and P. A. Pevzner. Whole genome duplications, multi-break rearrangements, and genome halving problem. In *SODA*, pages 665–679, 2007.
4. D. A. Bader, B. M. E. Moret, and M. Yan. A linear-time algorithm for computing inversion distance between signed permutations with an experimental study. *J. Comp. Biol.*, 8(5):483–491, 2001.

5. A. Bergeron, J. Mixtacki, and J. Stoye. A unifying view of genome rearrangements. In *Proceedings of WABI 2006*, volume 4175 of *LNBI*, pages 163–173. Springer Verlag, 2006.
6. A. Bergeron, J. Mixtacki, and J. Stoye. HP distance via double cut and join distance. In *Proceedings of CPM 2008*, pages 56–68, 2008.
7. A. Bergeron, J. Mixtacki, and J. Stoye. On computing the breakpoint reuse rate in rearrangement scenarios. In *Proceedings of RECOMB-CG 2008*, volume 5267 of *LNBI*, pages 226–240. Springer Verlag, 2008.
8. D.S. Cohen and M. Blum. On the problem of sorting burnt pancakes. *Discr. Appl. Math.*, 61(2):105–120, 1995.
9. T. Dobzhansky and A. H. Sturtevant. Inversions in the chromosomes of *Drosophila Pseudoobscura*. *Genetics*, 23:28–64, 1938.
10. P. Feijão and J. Meidanis. SCJ: A novel rearrangement operation for which sorting, genome median and genome halving problems are easy. In *Proceedings of WABI 2009, to appear*. Springer Verlag, 2009.
11. W. Gates and C. Papadimitiou. Bounds for sorting by prefix reversals. *Discr. Math*, 27:47–57, 1979.
12. S. Hannenhalli and P. A. Pevzner. Transforming men into mice (polynomial algorithm for genomic distance problem). In *Proceedings of FOCS 1995*, pages 581–592. IEEE Press, 1995.
13. S. Hannenhalli and P. A. Pevzner. Transforming cabbage into turnip: Polynomial algorithm for sorting signed permutations by reversals. *J. ACM*, 46(1):1–27, 1999. (First appeared in STOC 1995 Proceedings).
14. G. Jean and M. Nikolski. Genome rearrangements: a correct algorithm for optimal capping. *Inf. Process. Lett.*, 104:14–20, 2007.
15. Y. Lin and B. M. E. Moret. Estimating true evolutionary distances under the DCJ model. *Bioinformatics*, 24:i114–i122, 2008. (Proceedings of ISMB 2008).
16. J. Ma, L. Zhang, B. B. Suh, B. J. Raney, R. C. Burhans, W. J. Kent, M. Blanchette, D. Haussler, and W. Miller. Reconstructing contiguous regions of an ancestral genome. *Genome Research*, 16(12):1557–1565, 2006.
17. J. Meidanis, M. E. M. T. Walter, and Z. Dias. Reversal distance of signed circular chromosomes. In *Technical Report IC–00-23*. Institute of Computing, University of Campinas, 2000.
18. J. Mixtacki. Genome halving under DCJ revisited. In X. Hu and J. Wang, editors, *Proceedings of COCOON 2008*, volume 5092 of *LNCS*, pages 276–286. Springer Verlag, 2008.
19. J. H. Nadeau and B. A. Taylor. Lengths of chromosomal segments conserved since divergence of man and mouse. *Proc. Natl. Acad. Sci. USA*, 81:814–818, 1984.
20. M. Ozery-Flato and R. Shamir. Two notes on genome rearrangements. *J. Bioinf. Comput. Biol.*, 1(1):71–94, 2003.
21. P. A. Pevzner and G. Tesler. Transforming men into mice: the Nadeau-Taylor chromosomal breakage model revisited. In *Proceedings of RECOMB 2003*, pages 247–256, 2003.
22. D. Sankoff. Edit distances for genome comparison based on non-local operations. In *Proceedings of CPM 1992*, volume 644 of *LNCS*, pages 121–135, 1992.
23. D. Sankoff, R. Cedergren, and Y. Abel. Genomic divergence through gene rearrangement. In R. F. Doolittle, editor, *Molecular Evolution: Computer Analysis of Protein and Nucleic Acid Sequences*, volume 183 of *Meth. Enzymol.*, chapter 26, pages 428–438. Academic Press, San Diego, CA, 1990.
24. D. Sankoff and P. Trinh. Chromosomal breakpoint reuse in genome sequence rearrangement. *J. of Comput. Biol.*, 12(6):812–821, 2005.
25. E. Tannier, A. Bergeron, and M.-F. Sagot. Advances on sorting by reversals. *Discr. Appl. Math.*, 155(6-7):881–888, 2007.
26. G. Tesler. Efficient algorithms for multichromosomal genome rearrangements. *J. Comput. Syst. Sci.*, 65(3):587–609, 2002.
27. S. Yancopoulos, O. Attie, and R. Friedberg. Efficient sorting of genomic permutations by translocation, inversion and block interchange. *Bioinformatics*, 21(16):3340–3346, 2005.