Soft Computing manuscript No.
(will be inserted by the editor)

Computational Modelling of Interruptional Activities between Transposable Elements using Grammars and the Linear Ordering Problem

Lingling Jin · Ian McQuillan

Received: date / Accepted: date

Abstract Transposable elements (TEs) are DNA sequences that can either move or copy themselves to new positions within a genome. They constitute approximately 45% of the human genome. Knowing the evolution of TEs is helpful in understanding the activities of these elements and their impacts on genomes. In this paper, we devise a formal model providing notations/definitions that are compatible with biological nomenclature, while still providing a suitable formal foundation for computational analysis. We define sequential interruptions between TEs that occur in a genomic sequence to estimate how often TEs interrupt other TEs, useful in predicting their ages. We also describe the problem in terms of a matrix problem - the linear ordering problem. We then define the recursive interruption context-free grammar to capture the recursive nature in which TEs nest themselves into other TEs, and associate probabilities to convert the context-free grammar into a stochastic context-free grammar, as well as discuss how to use the CYK algorithm to find a most likely parse tree predicting TE nesting. We also discuss improvements on the theoretical model and adjust the parse trees to capture both sequential and recursive interruptional activities between TEs, and obtain more standard evolutionary trees.

Keywords transposable elements · formal modelling · interruptional analysis · linear ordering problem · stochastic context-free grammars

This research was supported by a grant from the Natural Sciences and Engineering Research Council of Canada.

Published in Soft Computing (2016) 20: 19–35, DOI: https://doi.org/10.1007/s00500-015-1725-2

Department of Computer Science,
University of Saskatchewan,
Saskatoon, SK, Canada

E-mail: lingling.jin@usask.ca

E-mail: mcquillan@cs.usask.ca

1 Introduction

In humans, coding sequences comprise less than 5% of the genome, whereas 66% to 69% of the genome is repetitive or repeat-derived [19], the majority of which are *transposable elements* (TEs), or *transposons*. TEs were first discovered by McClintock in 1949 as the genetic agents that are responsible for the sectors of altered pigmentation on mutant *Zea mays* kernels [28]. They are interspersed DNA sequences that can move or transpose themselves to new positions within the genome. TEs are found in nearly all species (both prokaryotes and eukaryotes, such as bacteria, fungi, plants and animals) that have been studied and constitute a large fraction of some genomes [12]. Depending on the organism, the proportion of TEs in the genome can differ widely, ranging from a few percent (3% in the yeast *Saccharomyces cerevisiae*) to a huge proportion encompassing almost the entire genome (>80% in maize). In particularly, the human genome is rich in TEs, at about 45% of the genome [25].

1.1 Classification

According to their mechanism of transposition, transposable elements are traditionally classified into two broad classes on the basis of their transposition mechanism and sequence organization [8]. Class I elements ("copy-and-paste" mechanism as the conceptual diagrams shown in Figure 1 (a)) are those that transpose via reverse transcription of an RNA intermediate, referred to as "*retrotransposons*". The RNA intermediate is transcribed from a genomic copy, then reverse-transcribed into DNA by a TE-encoded reverse transcriptase, and each complete replication cycle produces one new copy [32]. Consequently, retrotransposons rapidly increase the copy numbers of elements and thereby increase genome size. Class II elements ("cut-and-paste" mechanism as the conceptual

Table 1 The information of each type of transposable elements in the human genome.

TE Type	TE Class	Mode of Transposition	Length	Copy Number	Fraction of Genome
LINEs	Retrotransposons	Autonomous	6-8 kb	850,000	21%
SINEs	Retrotransposons	Non-autonomous	100-300 bp	1,500,000	13%
LTR retrotransposons	Retrotransposons	Autonomous	6-11 kb	450,000	8%
		Non-autonomous	1.5-3 kb		
DNA transposons	DNA transposons	Autonomous	2-3 kb	300,000	3%
		Non-autonomous	80-3,000 bp		

diagrams shown in Figure 1 (b)) which move predominantly via a DNA-mediated mechanism of excision and insertion, are often called "*DNA transposons*".

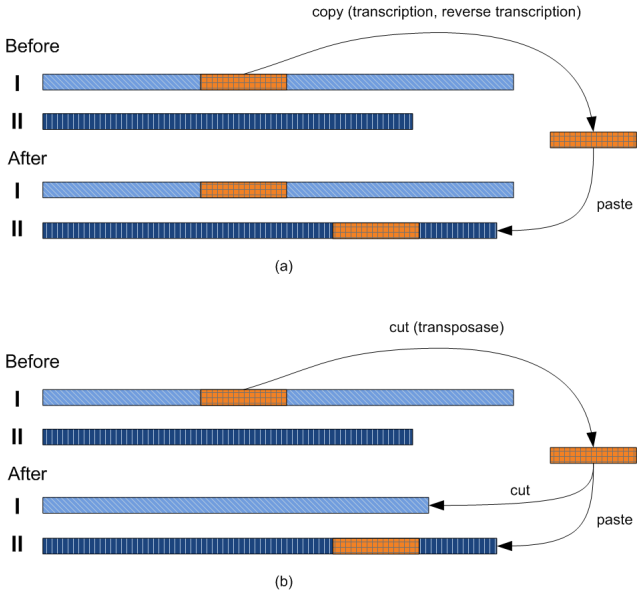

Fig. 1 Conceptual diagrams representing the transposition mechanisms. (a) Retrotransposons ("copy-and-paste" mechanism) copy themselves in two stages: first from DNA to RNA by transcription, then from RNA back to DNA by reverse transcription. The DNA copy is then inserted into the genome in a new position. (b) DNA transposons ("cut-and-paste" mechanism) do not involve an RNA intermediate. These transpositions are catalyzed by various types of transposase enzymes.

TEs are also described as being *autonomous* or *non-autonomous* based on whether or not they encode their own genes for transposition. Those transposable elements that possess a complete set of transposition protein domains are called *autonomous*. However, the term autonomous does not imply that an element is active or functional. Transposable elements that clearly lack an intact set of mobility-associated genes are called *non-autonomous* TEs, whose transposition requires participation of one or more proteins encoded by an autonomous element.

Within each of these classes, TEs can be further subdivided into several types on the basis of the structural features of their sequences. In general, the classification can be summarized as a tree structure in Figure 2, and the information of each type of TE is shown in Table 1 (information from [24]).

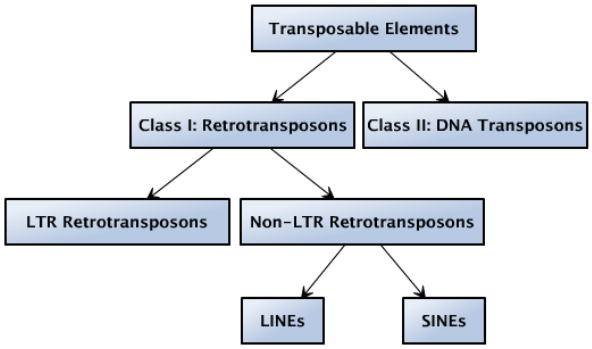

Fig. 2 The classification of transposable elements can be represented as a tree.

1.2 Interruptional Activities

Each transposable element has a distinct period of transpositional activity when it is active, in which it spreads through the genome, followed by inactivation and accumulation of mutations (they can also mutate while active). Though about half of the human genome derives from transposable elements, there has been a marked decline in their activities in the hominid lineage. DNA transposons appear to have become completely inactive and LTR retrotransposons may also have done so [24]. Because inactive TEs are so common in many genomes, throughout evolution, newer TEs end up nesting recursively, often multiple levels deep, inside existing inactive TEs. The result of the transpositional activities, over eons, can be described by (summarized from [10] and [22]):

1. older TEs are heavily interrupted by younger TEs, but have not inserted into younger elements;
2. younger TEs, with a relatively recent period of activity, have inserted into older elements that were present in the genome, but are not interrupted by older elements;
3. TEs of intermediate age have both inserted into older elements and been themselves fragmented by younger elements.

4. younger TEs interrupt not only older TEs, but also the fragmented TEs that had been previously interrupted. This makes the interruptions in the sequence even more complex. That is, interruptions are nested together recursively.

A novel method was introduced in [10] to estimate TE ages in mammalian genomes based on the frequencies with which every TE has inserted itself into every other TE. The resultant ordering that was obtained from a positional distribution agreed reasonably with published chronologies. This is in contrast to the more common divergence-based methods [17, 1, 30] to estimate TE ages, which has been unreliable, especially for older more diverged elements.

1.3 Motivations and our Contributions

The completion of the first human genome sequence [24] revealed that nearly half of our genome is derived from TEs, as shown in Table 1. The hundreds of millions of years of vertebrate evolution and the high density of TEs in our genome poses the question: what impact have they had on human evolution and human health? The composition and activities of human transposable elements have been reported to cause human diseases, including several types of cancer, such as breast cancer, colon cancer, retinoblastoma, neurofibromatosis, hepatoma, etc., through insertional mutagenesis of genes critical for preventing or driving malignant transformation [2]. Among all types of transposons, non-LTR retrotransposons are the predominant source of TE-related mutagenesis in the human genome. Thus, it is meaningful to understand the evolution of TEs and their impacts on the evolutions of the human genome, and the patterns of the activities of the TEs that may potentially cause human diseases.

In this paper, we will create a formal model capturing TE types and fragments within genomic sequences, called the *TE Fragment Model* in Section 2. Our model consists of initial definitions of TEs, the set of TEs, and the set of TE fragments. The model does not attempt to simulate or capture the molecular operations of TE movement (copying/cutting and pasting throughout a genome), which would create clear non-context-free patterns, making algorithmic analysis difficult. Rather, the model only describes the order and distance between TE fragments in genomic sequences by grouping homologous TEs together. At this level of abstraction, the model can be used to capture and calculate interruptions and their frequencies in a general way. We also give specialized definitions for use when this model is used with data from a prominent TE database called Repbase Update [16] and a common TE identification tool called RepeatMasker [31].

On top of the TE Fragment Model, we construct two separate models for different purposes. We will briefly discuss the method of estimating TE ages from [10] and calculate essentially the same data they used for their estimation, but using a specialized model called the *Sequential Interruptions Model* (Section 3), built on top of the TE Fragment Model. We will then associate our model to the *linear ordering problem*, a classic matrix optimization problem. This reduces the problem that the authors of [10] used to estimate TE ages to an existing well-studied problem in Section 4. Our second extension, the *Recursive Interruptions Model*, is done with a stochastic context-free grammar, used to capture recursive TE nesting. This allows for polynomial time parsing to be used to calculate a prediction of the nesting in Section 5. We will also discuss an approach to transform the grammar parse trees into evolutionary trees.

We attempt to make the models formal yet realistic and compatible with the biological literature on TEs. For this reason, the definitions are quite lengthy and make up a significant portion of this paper. However, we feel that it is necessary to contribute a suitable foundation for future computational analysis. Furthermore, multiple extensions and problems can be addressed on top of the same TE Fragment Model.

2 The TE Fragment Model

The purpose of this section is to develop a formal model of TEs and fragments of TEs in order to describe the biological concepts and problems clearly. It will be the starting point in which multiple other problems will be studied. We assume knowledge with context-free grammars [14] and parse trees, as well as common bioinformatics algorithms [15], such as pairwise and multiple alignments, and consensus sequences.

As a large part of research on bioinformatics is based on the analysis of DNA or amino-acid sequences, we will first briefly define a general sequence/string and other mathematical preliminaries in Definition 1.

Definition 1 We define several terms and notations:

An *alphabet* Σ is an abstract and finite set of symbols.

A *string* is any finite sequence of characters over an alphabet.

The *length* of s, denoted by $|s|$, is the number of characters in the string.

The *empty word* is denoted by λ and is of length 0.

The *set of all strings* (including the empty word) over Σ is denoted by Σ^*.

Let Σ be an alphabet and $s = s_1 s_2 \dots s_n$ be a string, $s_i \in \Sigma$, $1 \leq i \leq n$. Let j,k satisfy $1 \leq j \leq k \leq n$, then the *substring* of s which begins at the jth character, and ends at the kth character is

$$s(j,k) = s_j s_{j+1} \dots s_k.$$

Moreover, $s(j) = s_j$, is the jth character alone.

Let $s \in \Sigma^*$, then $frag(s)$ is the set of all possible *fragments (substrings)* of s. That is

$$frag(s) = \{s(p,q) \mid 1 \leq p \leq q \leq |s|\} \cup \{\lambda\}.$$

We extend this to sets of strings $S \subseteq \Sigma^*$ by

$$frag(S) = \bigcup_{s \in S} frag(s).$$

Given a set X, then $|X|$ is the *number of elements* in X.

When talking about a transposable element, life scientists usually are referring to a set of similar sequences that evolved from a single TE sequence. Therefore, we define a *transposable element* to itself be a set of strings (usually these strings will be similar to each other). We also define a *set of TEs*, an *instance of a TE* and the *consensus TE* in Definition 2.

Definition 2 A *transposable element (TE) X* is a finite set of strings (usually similar to each other) with $X \subseteq \Sigma^*$.

An *instance of a TE X* is an element $x \in X$.

A *consensus TE* is a consensus sequence of the elements of X.

A *set of TEs* χ is a finite set of TEs. That is, $\chi \subseteq 2^{\Sigma^*}$, χ is finite and each element of χ is finite.

Most of representative eukaryotic repetitive sequences have been compiled and reconstructed in a database called Repbase Update [16], which is a comprehensive database of the consensus sequences of repetitive elements (not only TEs, but also other repeats), that are present in diverse eukaryotic organisms.

Because of different biological contexts, it is also possible to interpret the set of TEs, χ, in multiple ways depending on the purpose. For example, we could use as χ the set of all TEs and TE instances that are present in a single genome, or as the set of sequences collected in Repbase Update, or any set of sequences that are all similar to the consensus sequences in Repbase Update within a threshold.

Knowing that one transposable element contains a number of instances, and each instance itself is a string, now we will define a *TE fragments set*. We expect to see many such fragments scattered throughout genomes as TEs become fragmented within a genome as they become interrupted by other TEs.

Definition 3 Let χ be a finite set of TEs. Then we call $\bar{\chi} \subseteq 2^{\Sigma^*}$ a *TE fragments set*, if for each element $\bar{X} \in \bar{\chi}$, there exists $X \in \chi$ such that $\bar{X}$ is a subset of $frag(X)$.

Thus, after picking a set of TEs, a TE fragments set is any set where each element consists of fragments of one transposable element in the set of TEs (separate elements in $\bar{\chi}$ could contain fragments from different TEs). Then in principle, we can pick any number of fragment sets for one set of TEs. For example, if we pick for χ to be the set of all TEs in the human genome, then we could have a TE fragments set $\bar{\chi}$ where each element contains fragments of separate TEs of length at least 50 (in this case, $\bar{\chi} = \{\bar{X} | x \in \bar{X}$ implies $|x| = 50$, $\bar{X} \subseteq frag(X)$, $X \in \chi$, for some $X\}$).

Although we defined TE fragments sets in a general way, we would also like to create a restriction to transposable elements that occur in present-day sequences. RepeatMasker (RM) [31] is the predominant library-based tool used in repeat identification, which has become a standard tool for any search of repeats in genomes. It is a sophisticated program that uses precompiled repeat libraries to find copies of known repeats represented in the libraries. The program performs a similarity search on both the "+" and "-" DNA strands based on local alignments, then outputs masked genomic DNA and provides a tabular summary of repeat content detected in both DNA strands. In the following definitions, we connect our general Definition 3 to the output of RepeatMasker.

Definition 4 Let s be a string representing some genomic sequence and χ_s be the set of TEs existing in s, then $\bar{\chi}(s \overset{\text{RM}}{\longleftrightarrow} \chi_s)$ is a *RepeatMasker TE fragments set*, running the program with a set of consensus TEs χ_s, against the genomic sequence s.

In other words, each element of $\bar{\chi}(s \overset{\text{RM}}{\longleftrightarrow} \chi_s)$ is a subset of some element of $\bar{\chi}$, where only TE fragments detected by the RepeatMasker program are selected.

For each TE fragment z of some TE X, and some $\bar{X} \in \bar{\chi}(s \overset{\text{RM}}{\longleftrightarrow} \chi_s)$, we associate a tuple in Definition 5, whose attributes are referred to in our model, which is also consistent with the output of the RepeatMasker program.

Definition 5 Given a genomic sequence s and a set of TEs χ_s, each *TE fragment z* in each $\bar{X} \in \bar{\chi}(s \overset{\text{RM}}{\longleftrightarrow} \chi_s)$ is a tuple:

$$\begin{aligned} info(z) = &(genoName, genoStart, genoEnd, \\ &genoLeft, strand, TEName, TEClass, \\ &TEStart, TEEnd, TELeft). \end{aligned} \tag{1}$$

We use the operator "." to access the attributes. For example, $z.TEname$ is the name of the TE to which fragment z belongs. The definition of each attribute is summarized in the list as follows (from [31]), and described in Example 1.

genoName: The name of the genomic sequence, where the fragment was detected.

genoStart: The start position of the fragment in the genomic sequence.

genoEnd: The end position of the fragment in the genomic sequence.

genoLeft: The opposite number of bases after the fragment in the genomic sequence.

Table 2 A table of two TE fragments in human chromosome 1.

Fragment	*genoName*	*genoStart*	*genoEnd*	*genoLeft*	*strand*	*TEName*	*TEClass*	*TEStart*	*TEEnd*	*TELeft*
z_1	chr1	430182	430604	-248820017	-	MER4E1	LTR	0	781	334
z_2	chr1	101403	101690	-249148931	+	AluJr	SINE	18	298	-14

strand: Relative orientation: "+" or "-".

TEName: The name of the TE to which the fragment belongs.

TEClass: The class of the TE to which the fragment belongs.

TEStart: The start position of the fragment in the TE consensus sequence to which the fragment belongs, if strand is "+"; or the opposite number of bases after the fragment in the TE consensus sequence, if strand is "-".

TEEnd: The end position of the fragment in the TE consensus sequence.

TELeft: The opposite number of bases after the fragment in TE consensus sequence, if the strand is "+"; or the start position of the fragment in the TE consensus sequence, if the strand is "-".

Then, two TE fragments in the same set of TEs, $\bar{X} \in \bar{\chi}(s \overset{\text{RM}}{\longleftrightarrow} \chi_s)$, have the same *genoName*. Also, a TE fragment can be present in either "+" or "-" strand, however, in both cases, we use the "+" strand coordinate to represent the location where it occurs. The orientations of the TE fragment are distinguished by the $TEStart$ or $TELeft$ attributes. For example, given a fragment z,

$$\begin{cases} z.TEStart \geq 0 \text{ and } z.TELeft \leq 0, \\ \qquad \text{if a fragment } z \text{ is in the "+" strand,} \\ z.TEStart \leq 0 \text{ and } z.TELeft \geq 0, \\ \qquad \text{if a fragment } z \text{ is in the "-" strand.} \end{cases}$$

In general, no matter in which strand a TE fragment occurs, our notations in the formal model are consistent. Example 1 picks two TE fragments showing the meanings of their attributes visually with respect to a genomic sequence and TE consensus sequences.

Example 1 Compare the Human Genome[1] chromosome 1, s, against the library of human transposable elements in Repbase Update, χ_s. The two TE fragments, z_1 and z_2, taken from two separate sets in the RepeatMasker TE fragments set, $\bar{\chi}(s \overset{\text{RM}}{\longleftrightarrow} \chi_s)$, are as listed in Table 2 with their detailed attributes.

Fig. 3 (a) shows a fragment of the transposon $MER4E1$, which was detected in the "-" strand of chromosome 1, and Fig. 3 (b) shows a fragment of the transposon $AluJr$, which was detected in the "+" strand of chromosome 1. Since the two fragments are detected in different strands, they are oriented oppositely as in Fig. 3.

[1] hg19, the Feb. 2009 assembly of the human genome.

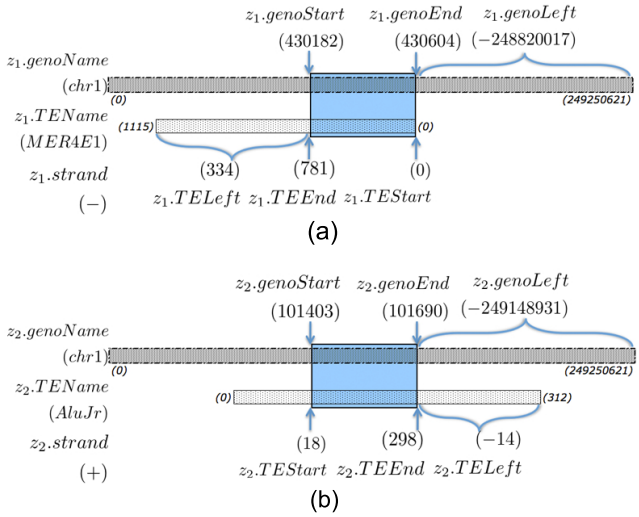

Fig. 3 The conceptual visualization of the TE fragments (in blue shadow) in Table 2. (a) visualizes the fragment z_1; (b) visualizes the fragment z_2. Note that the lengths of the visualized sequences in the figure are not proportional to their actual lengths.

Most of the present-day copies of TEs are detected by locally aligning the consensus TE sequences against a DNA sequence, thus the DNA sequence is fragmented into segments by the local aligned fragments. Some segments are detected as fragments of those TEs, while some are non-transposon DNA sequence. We then prune this DNA sequence to present only the TE segments. This process is defined in Definition 6.

Definition 6 Let s be a genomic sequence, χ_s a fixed ordering of the set of TEs in s, where $\chi_s = \{X_1, \ldots, X_m\}$, and $\bar{\chi}_s$ is a set of TE fragments. Assume $s = w_0 z_1 w_1 z_2 w_2\ z_3 \ldots z_k w_k$, with $z_1, \ldots, z_k$ in sets in $\bar{\chi}_s$, and no fragment of $w_0, w_1, \ldots, w_k$ are in sets in $\bar{\chi}_s$. Then a *pruned sequence* $\bar{s}$ of s with respect to $\bar{\chi}_s$ is

$$\bar{s} = \beta_0 z_1 \beta_1 z_2 \beta_2 \ldots z_k \beta_k, \text{ where } \beta_i = |w_i|,\ 0 \leq i \leq k. \tag{2}$$

That is, in a pruned sequence, we replace all non-TE fragments with their length.

In addition, from $\bar{s}$ and χ_s, we define an *order pruned sequence* $\bar{s}_o$ of $\bar{s}$ as the string over $\{1, \ldots, m\}^*$,

$$\bar{s}_o = j_1 j_2 \ldots j_k, \text{ where } z_i \in X_{j_i}, \text{ for all } i,\ 1 \leq i \leq k. \tag{3}$$

We can also extend a pruned sequence to a set of pruned sequences. Let $S = \{s_1, \ldots, s_N\}$, then the *set of pruned sequences* of S is $\bar{S} = \{\bar{s_1}, \ldots, \bar{s_N}\}$.

Table 3 A table of six TE fragments in human chromosome 1.

Fragment	*genoName*	*genoStart*	*genoEnd*	*genoLeft*	*strand*	*TEName*	*TEClass*	*TEStart*	*TEEnd*	*TELeft*
z_1	chr1	33632576	33632977	-215617644	+	L2a	LINE	2941	3379	-47
z_2	chr1	33633163	33633226	-215617395	+	L2b	LINE	3309	3374	-1
z_3	chr1	33633332	33633389	-215617232	-	MIR3	SINE	-19	189	128
z_4	chr1	33633467	33633769	-215616852	-	L2a	LINE	-2	3424	3074
z_5	chr1	33633802	33633941	-215616680	+	MLT1J	LTR	262	389	-123
z_6	chr1	33634011	33634148	-215616473	-	MER63A	DNA	-71	139	5

Example 2 is an example showing the pruned sequence and the order pruned sequence of a given genomic sequence segmented by the RepeatMasker detected TE fragments.

Example 2 A piece of the Human Genome chromosome 1 from position 33632576 to 33634148, s, is compared against the library of human transposable elements in Repbase Update, χ_s, where $\chi_s = \{X_1, X_2, X_3, X_4, X_5\}$, and the TE names of X_1, X_2, X_3, X_4, X_5 are *L2a*, *L2b*, *MIR*3, *MLT*1*J*, *MER*63*A*. The TE fragments taken from the RepeatMasker TE fragments set, $\bar{\chi}(s \overset{\text{RM}}{\longleftrightarrow} \chi_s)$, are as listed in Table 3 with their detailed attributes.

As in Definition 6, the genomic sequence s is

$$s = w_0 z_1 w_1 z_2 w_2 z_3 w_3 z_4 w_4 z_5 w_5 z_6 w_6.$$

This sequence is visualized in Fig. 4, where each fragment in the sequence is also marked as the order and the name of the TE, to which it belongs.

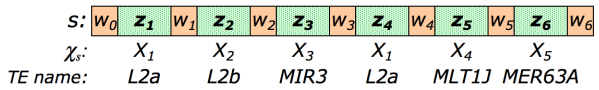

Fig. 4 The conceptual visualization of a genomic sequence (the Human Genome chromosome 1 from position 33632576 to 33634148), with the RepeatMasker detected TE fragments in Table 3. The TE fragments z_i, where $i = 1, \ldots, 6$ in the sequence are also marked with the notation of TEs $X_j \in \chi_s$, where $j = 1, \ldots, 5$, and the TE names to which they belong. Note that the lengths of the visualized sequences in the figure are not proportional to their actual lengths.

From Definition 6, the pruned sequence of s is

$$\bar{s} = \beta_0 z_1 \beta_1 z_2 \ldots z_6 \beta_6, \text{ where } \beta_i = |w_i|,\ 0 \leq i \leq 6,$$

and the order pruned sequence of s is

$$\bar{s_o} = 1\ 2\ 3\ 1\ 4\ 5, \text{ where } z_i \in X_{j_i}, \text{ for all } i,\ 1 \leq i \leq 6.$$

So far, we have defined some fundamental concepts associated with key biological terms, such as a transposable element, a TE fragment, and a pruned sequence, and also extended them to sets within what we call the TE Fragment Model. In the next section, we will move the emphasis to the first of two uses of the model, to describe the dynamic interruptional activities between different TEs.

3 The Sequential Interruption Model

As discussed in Section 1.2, newer TEs tend to interrupt older TEs, thereby fragmenting older TEs within the single linear sequence. By analyzing that sequence, we are able to predict where and how often the insertional and transpositional activities occurred throughout evolution, which is useful information in predicting an order that those activities occurred, and further, potentially inferring the ages of these TEs.

We can capture these interruptional activities with our model in this section. We will first describe and define *sequential interruptions*, then the *interruption matrix*. We will also connect our model to the *Linear Ordering Problem* in Section 4, whose methods can be used in solving our problem. We will also discuss other applications of our model throughout this section to Section 5.

To analyze interruptional patterns, we are only interested in TE fragments and their relative positions in a genomic sequence. In [10], they classify an interruption as occurring when one TE fragment is within a certain distance from a fragment on the left and a fragment on the right, where both are from the same TE, and the two fragments are "close to" continuous within the TE. They then compile this information into a so-called *adjacency matrix*, or interruptional matrix, giving an estimate on the number of times each TE interrupted each other. We can calculate this same analysis using the TE Fragment Model, and in particular, pruned sequences in Definition 6. This definition provides all that is necessary to calculate our interruptional matrix. Before defining a sequential interruption in Definition 8, we need to define *continuous TE fragments* first.

Definition 7 Let s be a genomic sequence with a set of TEs χ_s, TE fragment set $\bar{\chi}(s \overset{\text{RM}}{\longleftrightarrow} \chi_s)$ and pruned sequence $\bar{s} = \beta_0 z_1 \beta_1 z_2 \ldots z_k \beta_k$ as in Equation (2). Then two TE fragments z_i and z_j $(i < j)$ are *continuous TE fragments*, $z_i \overset{\varepsilon,E}{\sim} z_j$, with distance $\varepsilon \in \mathbb{N}$ (in the consensus sequence) and distance $E \in \mathbb{N}$ (in the genomic sequence), if they satisfy the following conditions:

1. they belong to the same transposable element:

$$z_i.TEName = z_j.TEName$$

2. they are detected in the same DNA strand:

$$z_i.strand = z_j.strand$$

3. they are either separated or overlap [2] with less than or equal to a distance, ε, with respect to the TE consensus sequence to which they belong:

$$\begin{cases} abs(z_j.TE\ Start - z_i.TEEnd) \leq \varepsilon, \\ \qquad \text{if } z_i \text{ and } z_j \text{ occur in the "+" strand.} \\ abs(z_i.TE\ \ Start - z_j.TEEnd) \leq \varepsilon, \\ \qquad \text{if } z_i \text{ and } z_j \text{ occur in the "-" strand.} \end{cases}$$

4. They are in the genomic sequence within a distance, E, of non-transposon DNA sequence:

$$\sum_{r=i}^{j-1} \beta_r \leq E.$$

Notice that continuous TE fragments are not necessarily beside each other in the genomic sequence, as there can be a distance of E between them. Some continuous TE fragments appear to have an overlap of duplication of a portion of the transposon. This is because RepeatMasker often extends the homology match of both fragments to the TE consensus sequence by several base pairs.

Definition 8 Given a genomic sequence s, a set of TEs with a fixed ordering on its elements $\chi_s = \{X_1, X_2, \ldots, X_m\}$, and a distance $\varepsilon \in \mathbb{N}$ in TE consensus sequence, a distance $E \in \mathbb{N}$ in genomic sequence as in Definition 7, as well as a pruned sequence $\bar{s} = \beta_0 z_1 \beta_1 z_2 \ldots z_k \beta_k$, as in Equation (2). We define *sequential interruptions* of X_j by X_i as

$$\begin{aligned} \Xi_s^{\varepsilon,E}(X_i, X_j) = &\{k \mid z_k \in \bar{X}_i, z_{k-\eta_1}, z_{k+\eta_2} \in \bar{X}_j, \\ &z_{k-\eta_1} \overset{\varepsilon,E}{\sim} z_{k+\eta_2}, \text{ and } \eta_1, \eta_2 \in \mathbb{N}\}. \end{aligned} \tag{4}$$

Thus X_i is called *interrupter*, and X_j is called *interruptee*.

The frequencies with which the interruptions between different TEs occur in the sequence can also infer the activities of these TEs. Therefore, we also define the abundance of interruptions to capture the frequencies of interruptions in Definition 9 to represent interruptions in a general way.

Definition 9 Given a genomic sequence s, a set of TEs with a fixed ordering on its elements $\chi_s = \{X_1, X_2, \ldots, X_m\}$, the *abundance that X_i interrupts X_j* in s, $1 \leq i \leq m$, $1 \leq j \leq m$, is defined as the total number of times that X_i interrupts X_j. The abundance is equal to

$$|\Xi_s^{\varepsilon,E}(X_i, X_j)|.$$

For the genome S that has chromosomes $s_1, s_2, \ldots, s_N$, we then add up the abundance that X_i interrupts X_j for all chromosomes, which is

$$|\Xi_S^{\varepsilon,E}(X_i, X_j)| = \sum_{n=1}^{N} |\Xi_{s_n}^{\varepsilon,E}(X_i, X_j)|.$$

[2] We calculate the amount that separate them or the amount they overlap using the *abs*() function to get the absolute value.

The *interruption array* of X_i on S, for $1 \leq i \leq m$, is the array

$$M(i) = [|\Xi_S^{\varepsilon,E}(X_i, X_j)|]_{j=1,\ldots,m}.$$

The *interruption matrix* on S is an $m \times m$ matrix defined by

$$M = [|\Xi_S^{\varepsilon,E}(X_i, X_j)|]_{\substack{i=1,\ldots,m \\ j=1,\ldots,m}}.$$

The interruption array and matrix are different ways to structure the abundance by using the ordering on the elements in χ_s. This interruption matrix was calculated in much the same way as the adjacency matrix of [10].

In Example 3, we illustrate how to apply the model of sequential interruptions in a real situation to find sequential interruptions, and calculate the interruptional matrix.

Example 3 Table 4 is a list of five TE fragments from chromosome 1 position 448062 to 449273 taken from the RepeatMasker TE fragments set, $\bar{\chi}(s \overset{\text{RM}}{\longleftrightarrow} \chi_s)$, in Example 1. The five fragments belong to three TEs: X_1, X_2 and X_3, where the TE names of X_1, X_2, X_3 are $L1MD3$, $AluYc$, $AluSq$.

As in Definition 6, the genomic sequence s is

$$s = w_0 z_1 w_1 z_2 w_2 z_3 w_3 z_4 w_4 z_5 w_5,$$

as visualized in Fig. 5.

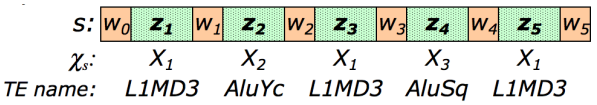

Fig. 5 The conceptual visualization of a genomic sequence (the Human Genome chromosome 1 from position 448062 to 449273), with the RepeatMasker detected TE fragments in Table 4. Note that the lengths of the visualized sequences in the figure are not proportional to their actual lengths.

The pruned sequence of s is

$$\bar{s} = \beta_0 z_1 \beta_1 z_2 \beta_2 z_3 \beta_3 z_4 \beta_4 z_5 \beta_5,$$

where $z_1, z_3, z_5 \in \bar{X}_1$, $z_2 \in \bar{X}_2$, $z_4 \in \bar{X}_3$, $\bar{X}_1, \bar{X}_2, \bar{X}_3 \in \bar{\chi}(s \overset{\text{RM}}{\longleftrightarrow} \chi_s)$, $\beta_0, \ldots, \beta_5 \in \mathbb{N}$, and $z_1 \overset{\varepsilon,E}{\sim} z_3$, $z_3 \overset{\varepsilon,E}{\sim} z_5$, as shown in Fig. 5.

It is possible to see that there are two potential interruptions in s: an instance of X_1 is present in the sequence, then an instance of X_2 and an instance of X_3 potentially inserted themselves into the instance of X_1 to break it into three segments z_1, z_3 and z_5; that is, $|\Xi_s^{\varepsilon,E}(X_2, X_1)| = 1$ and $|\Xi_s^{\varepsilon,E}(X_3, X_1)| = 1$.

Given a fixed order of the set of TEs as

$$\chi_s = \{\ldots, X_1, \ldots, X_2, \ldots, X_3, \ldots\},$$

Table 4 An example of sequential interruptions in chromosome 1.

Fragment	*genoName*	*genoStart*	*genoEnd*	*genoLeft*	*strand*	*TEName*	*TEClass*	*TEStart*	*TEEnd*	*TELeft*
z_1	chr1	448062	448139	-248802482	+	L1MD3	LINE	6988	7068	-814
z_2	chr1	448150	448328	-248802293	+	AluYc	SINE	122	299	0
z_3	chr1	448332	448403	-248802218	+	L1MD3	LINE	7068	7148	-847
z_4	chr1	448403	448710	-248801911	+	AluSq	SINE	1	313	0
z_5	chr1	448710	449273	-248801348	+	L1MD3	LINE	7149	7753	-242

the interruption matrix showing only the rows and columns of these TEs is

$$M = \begin{bmatrix} \vdots & \vdots & \vdots \\ \dots\ 0 \dots & 0 \dots & 0 \dots \\ \vdots & \vdots & \vdots \\ \dots\ 1 \dots & 0 \dots & 0 \dots \\ \vdots & \vdots & \vdots \\ \dots\ 1 \dots & 0 \dots & 0 \dots \\ \vdots & \vdots & \vdots \end{bmatrix}.$$

From the analysis on these sequential interruptions, we can predict that the age of $L1MD3$ might be older than both $AluYc$ and $AluSq$, but this provides no clue as to which one of $AluYc$ and $AluSq$ is older, because we do not know which of the two independent interruptions occurred first.

Using the operations of matrix rearrangement in Section 4, we can also predict a potential chronology of these TEs, by reducing the problem employed in [10] to an existing matrix optimization problem.

Our notions in this section transformed the interruptional matrix construction described in prose in [10] into a formal model, which is more clear, and can also be used for other purposes, such as the study of recursive patterns, as we will do in Section 5.

4 Linear Ordering Problem for Sequential Interruptions Analysis

The interruptional analysis done in [10] was performed by using the interruptions between TEs, then rearranged the TEs in such a way that they hypothesized would order them from oldest to youngest. They calculated an adjacency matrix comparable to an interruption matrix in Definition 9, whose rows/columns correspond to a TE ordering, which counts the number of interruptions between each pair of TEs. They then used algorithms of a complexity of $O(n!)$, where n is the number of TEs, to predict the relative age order of TEs by rearranging the rows and columns of the matrix to achieve a lowest penalty score (the summation of nonzero entries in the upper triangle of the matrix), corresponding to reordering the TEs, from those that get interrupted most while interrupting least, to those that interrupt most while getting interrupted least. So that in a hypothetical matrix, the TEs are arranged in a predictive chronological order of decreasing in age (from oldest to youngest).

Our formal model of sequential interruptions from Section 3 calculated an interruption matrix. Next, we will map this matrix to the linear ordering problem, which rearranges a matrix similar as the approach in [10]. First of all, we will examine a set of matrix rearrangement operations in linear algebra, in order to describe and compute the linear ordering problem.

When working with rearranging some objects or values, the act of rearrangement is a permutation as defined in Definition 10.

Definition 10 A *permutation* π is a bijective function from an ordering of n elements $(1\ 2\ 3\ \cdots\ n)$ to itself. It will be denoted by an n-tuple where the number at position i is $\pi(i)$. The $\pi(i)$ gives the position of element i in the new ordering.

A *permutation matrix* is a square $n \times n$ binary matrix that has exactly one entry 1 in each row and each column and 0s elsewhere. Specifically, the permutation matrix of π is a matrix P_π whose entries are all 0 except that in row i, the entry $\pi(i)$ equals 1.

Each such matrix represents a specific permutation of n elements and, when multiplying another $n \times n$ matrix A with P from the left, it permutes the rows of A. Further, multiplying A with the transpose of P, P^T, from the right, permutes the columns of A.

Example 4 illustrates a permutation of an ordering, and its permutation matrix, as well as how to permute a square matrix using this permutation.

Example 4 For an ordering of $(1\ 2\ 3\ 4\ 5)$, a permutation could be $\pi = (1\ 4\ 2\ 5\ 3)$, where $\pi(1) = 1, \pi(2) = 4, \pi(3) = 2, \pi(4) = 5, \pi(5) = 3$.

The permutation matrix P_π of π is

$$P_\pi = \begin{bmatrix} 1\ 0\ 0\ 0\ 0 \\ 0\ 0\ 0\ 1\ 0 \\ 0\ 1\ 0\ 0\ 0 \\ 0\ 0\ 0\ 0\ 1 \\ 0\ 0\ 1\ 0\ 0 \end{bmatrix}.$$

As in Definition 10, any square matrix with n rows and columns can be rearranged by a permutation of n elements, using its permutation matrix.

Given another square matrix

$$A = \begin{bmatrix} 11 & 12 & 13 & 14 & 15 \\ 21 & 22 & 23 & 24 & 25 \\ 31 & 32 & 33 & 34 & 35 \\ 41 & 42 & 43 & 44 & 45 \\ 51 & 52 & 53 & 54 & 55 \end{bmatrix},$$

multiplying A with P_π from the left permutes the rows of A:

$$P_\pi \times A = \begin{bmatrix} 11 & 12 & 13 & 14 & 15 \\ 41 & 42 & 43 & 44 & 45 \\ 21 & 22 & 23 & 24 & 25 \\ 51 & 52 & 53 & 54 & 55 \\ 31 & 32 & 33 & 34 & 35 \end{bmatrix}.$$

While multiplying A with P_π^T from the right permutes the columns of A:

$$A \times P_\pi^T = \begin{bmatrix} 11 & 14 & 12 & 15 & 13 \\ 21 & 24 & 22 & 25 & 23 \\ 31 & 34 & 32 & 35 & 33 \\ 41 & 44 & 42 & 45 & 43 \\ 51 & 54 & 52 & 55 & 53 \end{bmatrix}.$$

So that $P_\pi \times A \times P_\pi^T$ permutes A with the permutation $\pi = \begin{pmatrix} 1 & 4 & 2 & 3 & 5 \end{pmatrix}$:

$$P_\pi \times A \times P_\pi^T = \begin{bmatrix} 11 & 14 & 12 & 15 & 13 \\ 41 & 44 & 42 & 45 & 43 \\ 21 & 24 & 22 & 25 & 23 \\ 51 & 54 & 52 & 55 & 53 \\ 31 & 34 & 32 & 35 & 33 \end{bmatrix}.$$

Permuting a square matrix is the operation used in the linear ordering problem in the next subsection.

4.1 Linear Ordering Problem

The linear ordering problem is one of the classical combinatorial optimization problems which was already classified as $\mathcal{NP}$-hard in 1979 by Garey and Johnson [9]. This problem is described in [29] as:

Given an $m \times m$ matrix C, the *linear ordering problem* is the problem of finding a permutation π of the column and row indices $\{1, \cdots, m\}$, such that the value

$$f(\pi) = \sum_{i=1}^{m} \sum_{j=i+1}^{m} c_{\pi(i),\pi(j)} \tag{5}$$

is maximized. In other words, the goal is to find a permutation of the columns and rows of C such that the sum of the elements in the upper triangle is maximized.

Analogically, the goal of the TE sequential interruptional analysis is to find a permutation of TE ordering that maximizes the sum of upper triangle of the interruption matrix in Definition 9. The sequential interruption analysis can be described in terms of the linear ordering problem as follows.

Given a set of genomic sequences, S, a set of TEs with a fixed ordering on its elements,

$$\chi_s = \{X_1, X_2, \ldots, X_m\},$$

and an interruption matrix of χ_s on S,

$$IM = [|\Xi_S^{\varepsilon,E}(X_i, X_j)|]_{\substack{i=1,\ldots,m \\ j=1,\ldots,m}},$$

the problem is to find a permutation π of χ_s, corresponding to the column and row indices $\{1, \cdots, m\}$, such that the value

$$f(\pi) = \sum_{i=1}^{m} \sum_{j=i+1}^{m} IM_{\pi(i),\pi(j)} \tag{6}$$

is maximized.

The resultant permutation of χ_s corresponds to a hypothetical chronological order of TEs in χ_s of increasing in age, as it is optimizing essentially the same function used to estimate the ages (in [10], they attempt to find a permutation, corresponding to TEs of decreasing in age, that minimizes the summation of nonzero entries in the upper triangle of the matrix). The resulting matrix whose rows and columns are rearranged, will have the following features (as shown in Fig. 6):

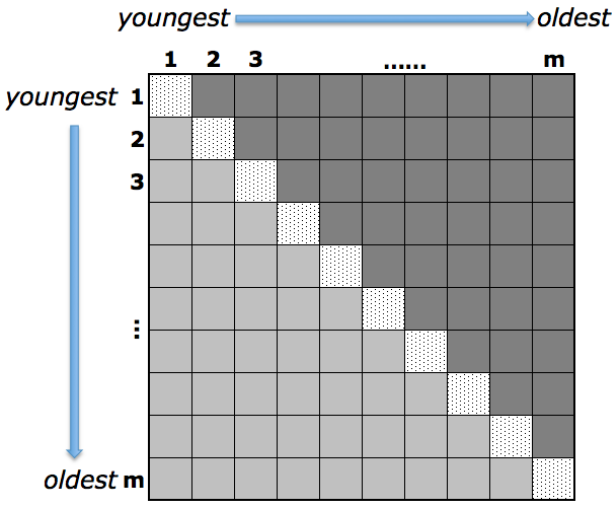

Fig. 6 A conceptual diagram of the permuted interruptional matrix with a maximum sum of upper triangle. The row and column of the matrix correspond to the resultant permutation of the TE ordering, which is a hypothetical chronological order of TEs of increasing in age.

1. Each item in the matrix records the number of interruptions that an interrupter (on the vertical axis) has inserted itself into an interruptee (on the horizontal axis);

2. The order of TEs on the vertical axis (from top to bottom) is the same as the order of TEs on the horizontal axis (from left to right), which is arranged in chronological order of increasing in age (from youngest to oldest).
3. The matrix can be divided to four portions:
   - the top-left portion of the matrix represents young TEs interrupting young TEs;
   - the top-right portion of the matrix represents young TEs interrupting old TEs;
   - the bottom-left portion of the matrix represents old TEs interrupting young TEs;
   - the bottom-right portion of the matrix represents old TEs interrupting old TEs.
4. In theory, the lower triangle region of the matrix (light grey in Fig. 6), corresponds to older TEs interrupting younger TEs, and should be mainly populated by zeros, meaning that there are no interruptions. Non-zeros in this region might occur because of defragmentation errors, or other mutation events that give the appearance of TE insertion.
5. Most non-zero values should appear in the upper triangle region of the matrix (dark grey in Fig. 6), which corresponds to young TEs interrupting old TEs.
6. Interruptions of the same type of TEs into themselves (which would be recorded directly on the matrix diagonal) are not scored due to the fact that they are difficult to confidently identify and do not affect the ordering analysis.

The linear ordering problem is $\mathscr{NP}$-hard, that is, we cannot expect to find a polynomial time algorithm for its solution. After computing an interruption matrix of n TEs using our model in Section 3, a straightforward method to find the permutation of the problem, would be exhaustive search: apply all $n!$ possible permutations to the interruption matrix and the resultant permutation will be the one with which the permuted interruption matrix achieves the maximum score over all $n!$ sum of upper triangle scores. The algorithm has a complexity of $O(n!)$, which is considerably inefficient.

In our experiment, there are around 900 ($n = 900$) human TEs taken from Repbase that are compared with the Human Genome. As such, it is not feasible to use the exhaustive search to find a permutation exactly. However, since the linear ordering problem arises in a variety of applications, algorithms for its efficient solution are required. There are some exact methods that use *Branch-and-Bound* algorithms to solve the problem to (proven) optimality discussed in [26].

For example, the branch-and-bound with partial orderings in [5], the lexicographic search algorithm in [20,21], and the branch-and-bound approach, where Lagrangian relaxation techniques are used for bound computations in [4]. The branch-and-bound can also be realized in a special way leading to the so-called *Branch-and-Cut* method, which is essentially a branch-and-bound algorithm, where the upper bounds are computed using linear programming relaxations as discussed in [26].

As opposed to exact methods, which guarantee to give an optimum solution of the problem, heuristic methods only attempt to yield a good, but not necessarily optimal solution. Nevertheless, the time taken by an exact method to find an optimum solution to a difficult problem, if indeed such a method exists, is in a much greater order of magnitude than the heuristic one. Thus we often resort to heuristic methods to solve real optimization problems. There are some heuristic algorithms summarized in [26] as well, such as GRASP [7], tabu search [11], the simulated annealing method [18], variable neighbourhood search [13], scatter search [23], iterated local search [3], etc. A computational comparison of some heuristic algorithms (including the above mentioned ones and some others) on a benchmark library done in [27], concluded that the iterated local search method achieved the best result among them. We leave an exhaustive comparison of these method on the interruption matrix as future work.

5 The Recursive Interruption Model

When many insertions occurred throughout the evolution of a genomic sequence, the interruptions nest in a recursive pattern [22], which cannot be represented entirely with the interruptional matrix that only counts the abundance without storing the hierarchical relationships of interruptions. Indeed, Example 5 shows some nested TEs in real data.

Example 5 Table 5 is a list of TE fragments taken from the RepeatMasker TE fragments set, $\bar{\chi}(s \overset{\text{RM}}{\longleftrightarrow} \chi_s)$, where s is the X chromosome of the Human Genome, and χ_s is the library of human transposable elements in Repbase Update. These seven TE fragments start from the X chromosome position 53437061 to 53438226 that belong to four TEs: X_1, X_2, X_3 and X_4, where the TE names of X_1, X_2, X_3, X_4 are MIR, $AluJb$, $AluSx$, $AluSq2$.

As in Definition 6, the genomic sequence s is

$$s = w_0 z_1 w_1 z_2 w_2 z_3 w_3 \dots z_7 w_7,$$

as visualized in Fig. 7.

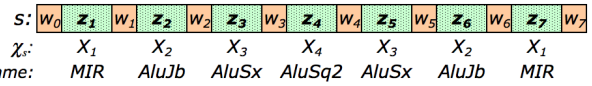

Fig. 7 The conceptual visualization of a genomic sequence (the Human Genome X chromosome from position 53437061 to 53438226), with the RepeatMasker detected TE fragments in Table 5. Note that the lengths of the visualized sequences in the figure are not proportional to their actual lengths.

The pruned sequence of s is

$$\bar{s} = \beta_0 z_1 \beta_1 z_2 \beta_2 z_3 \beta_3 z_4 \beta_4 z_5 \beta_5 z_6 \beta_6 z_7 \beta_7,$$

Table 5 An example of recursive interruptions in the X chromosome.

Fragment	*genoName*	*genoStart*	*genoEnd*	*genoLeft*	*strand*	*TEName*	*TEClass*	*TEStart*	*TEEnd*	*TELeft*
z_1	chrX	53437061	53437143	-101833417	+	MIR	SINE	3	88	-174
z_2	chrX	53437143	53437277	-101833283	+	AluJb	SINE	1	132	-170
z_3	chrX	53437277	53437448	-101833112	+	AluSx	SINE	39	192	-120
z_4	chrX	53437448	53437761	-101832799	+	AluSq2	SINE	1	312	0
z_5	chrX	53437761	53437887	-101832673	+	AluSx	SINE	193	312	0
z_6	chrX	53437887	53438055	-101832505	+	AluJb	SINE	133	293	-9
z_7	chrX	53438055	53438226	-101832334	+	MIR	SINE	89	261	-1

where $\beta_0,\ldots,\beta_7 \in \mathbb{N}$, $z_1,z_7 \in \bar{X}_1$, $z_2,z_6 \in \bar{X}_2$, $z_3,z_5 \in \bar{X}_3$, $z_4 \in \bar{X}_4, \bar{X}_1,\bar{X}_2,\bar{X}_3,\bar{X}_4 \in \bar{\chi}(s \overset{\text{RM}}{\longleftrightarrow} \chi_s)$, and $z_1 \overset{\varepsilon,E}{\sim} z_7$, $z_2 \overset{\varepsilon,E}{\sim} z_6$, $z_3 \overset{\varepsilon,E}{\sim} z_5$.

It is possible to see a potential process of nested interruptions described as:

- at first, an instance of *AluJb* inserted itself into an instance of *MIR* to break it into z_1 and z_7;
- then an instance of *AluSx* inserted itself into the instance of *AluJb* that has already presented in the sequence, to break it into z_2 and z_6;
- more recently, an instance of *AluSq2* (z_4) inserted itself into the presented *AluSx* instance to break it into z_3 and z_5.

From the interruption analysis above, we can predict from the recursive interruptions that the age order of these three TEs from oldest to youngest might be: *MIR*, *AluJb*, *AluSx*, *AluSq2*.

The nested nature of the interruptions in Example 5 is not captured by the interruptional matrix as done in Section 3, or in [10], because the recursive nesting can "push" fragments so that they are no longer continuous. However, these nested interruptions are very informative in predicting the chronological order of when these interruptions occurred in the genomic sequence. Therefore, we will create a new model built on top of the TE Fragment Model in this section to capture this hierarchical nesting feature. We will first define a *recursive interruption context-free grammar* to model the generation of recursive interruptions, then discuss algorithms that calculate a parse tree of the grammar generating a given *order pruned sequence*, which shows a prediction of the hierarchical structure of TE insertions. Afterwards, we will further discuss some modifications to the the parse tree representation in order to simplify and improve these trees to better discuss evolutionary trees.

5.1 Context-free Grammar to Generate Recursive Interruptions

We provide a theoretical model in this subsection to describe the nature of recursive interruptions using a context-free grammar.

Definition 11 Given a set of TEs with a fixed order on its elements, $\chi = \{X_1,X_2,\ldots,X_m\}$, the *recursive interruption context-free grammar* is a grammar $G = (V,T,\delta,S)$, where $V = \{S,X_1,X_2,\ldots,X_m\}$, $T = \{1,2,\ldots,m\}$, and δ contains the following productions:

$$S \to X_i S, \qquad 1 \le i \le m, \tag{1}$$
$$S \to X_i, \qquad 1 \le i \le m, \tag{2}$$
$$X_i \to X_i X_j X_i, \quad 1 \le i \le m,\ 1 \le j \le m, \tag{3}$$
$$X_i \to i, \qquad 1 \le i \le m. \tag{4}$$

This grammar is used to generate strings over $\{1,\ldots,m\}^*$ corresponding to TE orders. Intuitively, productions of type (3) correspond to an instance of X_j inserting itself throughout evolution into an instance of X_i, as shown in Fig. 8, leaving a fragment from i, then j, then i.

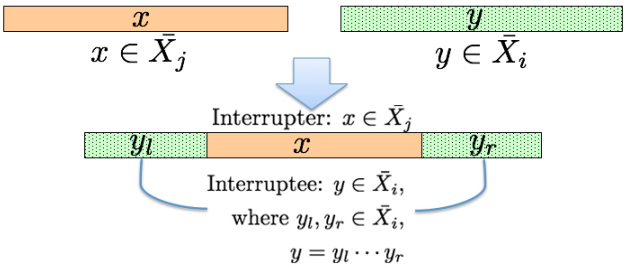

Fig. 8 A diagram showing an instance of X_j inserting itself throughout evolution into an instance of X_i, corresponding to an application of a production of type (3).

In a sentential form, $X_iX_jX_i$ can either derive iji (using productions of type (4)) corresponding to that order of TEs, or any of them can be further interrupted (using productions of type (3)). Productions of type (1) correspond to independent positions of the sequence where a TE can insert itself (not a nested insertion, and can only be produced continuously from the root along the rightmost path of a parse tree). Productions of type (2) correspond to the final independent position of a TE insertion.

This context-free grammar is ambiguous (meaning that multiple parse trees can give the same string). Indeed, it is clear that any string over T^+ can be generated by G by

using only productions of types (1), (2) and (4). This would require the application of $2k$ productions to generate a string of length k. However, for every application of a production of type (3), the total number of productions needed to generate a string of length k decreases. If there are l productions of type (3) applied, the total number of productions needed to generate a string of length k decreases to $2(k-l)$.

Since each production of type (1), (2), or (3) corresponds to one biological transposition, we are interested in parse trees which maximize the application of productions of type (3), or minimize the total number of productions applied. This would correspond to minimizing the number of transpositions that occurred throughout evolution.

Example 6 shows how nested interruptions in a sequence are generated by the grammar as the yield of its one possible parse tree that maximized the application of productions of type (3).

Example 6 Given a genomic sequence s and a set of TEs with a fixed order on its elements $\chi_s = \{X_1, X_2, \ldots, X_{10}\}$, and assume

$$s = w_0 z_1 w_1 \ldots z_{13} w_{13},$$

as in Equation (2), with $z_1, z_4, z_6 \in \bar{X_2}$, $z_2, z_8, z_{10}, z_{12} \in \bar{X_3}$, $z_5 \in \bar{X_4}$, $z_{11}, z_{13} \in \bar{X_5}$, $z_3, z_7 \in \bar{X_6}$, $z_9 \in \bar{X_{10}}$. Then an order pruned sequence

$$\bar{s_o} = 2\ 3\ 6\ 2\ 4\ 2\ 6\ 3\ 10\ 3\ 5\ 3\ 5$$

is the yield of the parse tree shown in Fig. 9.

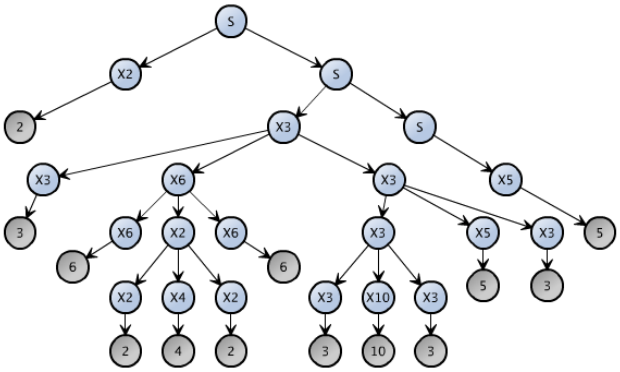

Fig. 9 A parse tree of G from Definition 11 that yields $\bar{s_o}$.

The recursive interruption context-free grammar in Definition 11 is a very simple and general way of capturing the recursive nature of TE interruptions. However, the order pruned sequence generated by the grammar only contains the TEs (names/order of TEs) to which the detected TE fragments belong. It does not take into account where each fragment lies within a TE with the current grammar. It is not clear how one could take the positional information into account to determine whether two TE fragments are continuous fragments (Definition 7), then further determine the existence of an interruption in an order pruned sequence.

5.2 Algorithms for Finding a Parse Tree

As discussed, given an order pruned sequence, we are interested in finding a parse tree of the grammar that maximizes the applications of productions of type (3), or minimizes the overall productions applied to generate this sequence. In this subsection, we will discuss some methods to find such parse trees by converting the recursive interruption context-free grammar into a stochastic context-free grammar.

A *stochastic context-free grammar* is indeed a context-free grammar, where every production in the grammar has an associated probability value between 0 and 1, such that the probability for all productions on a nonterminal adds to 1. The probability associated with a parse tree is the product of the probabilities of the production instances applied to produce it.

Considering the context-free grammar in Definition 11, since all probabilities are between 0 and 1, trees that use fewer productions will tend to have a higher probability. A *most likely parse tree*, defined as a parse tree with the highest probability, corresponds to the parse tree that has the most productions of type (3) applied in the recursive interruption context-free grammar. For this grammar, if we give all productions for each nonterminal equal weight (for each production of X_i, the probability is $1/(m+1)$, and for each production of S, the probability is $1/(2m)$), the CYK algorithm [6] can find a most likely parse tree that has a given sequence as yield. In our case, starting with the order pruned sequence, it can predict a most likely parse tree with it as the yield.

The complexity of the CYK algorithm is $O(L^3N^3)$ [6], where L is the length of the order pruned sequence (corresponding to the number of TE fragments detected in a genomic sequence), and N is the number of nonterminals in the grammar (corresponding to the total number of transposons of that organism in Repbase Update plus one), which will be very lengthy in practice.

We leave as future work an investigation of algorithms that can be efficient while taking additional positional information into account in generating the most likely parse tree.

5.3 Adjustments to the Parse Tree

In the grammar in Definition 11, the productions of type (1) and (3) determine the generation of interruptions from the left to the right side of the sequence. This places independent interruptions at differing heights of the parse tree - as

Table 6 An example of three atomic patterns of interruptions. Group (a) is a single interruption; group (b) shows two sequential interruptions; group (c) shows two recursive interruptions. Their corresponding trees are in Fig. 10.

group	*genoName*	*genoStart*	*genoEnd*	*genoLeft*	*strand*	*TEName*	*TEClass*	*TEStart*	*TEEnd*	*TELeft*
(a)	chr1	23803	24038	-249226583	+	L2b	LINE	2940	3212	-175
	chr1	24087	24250	-249226371	+	MIR	SINE	49	260	-2
	chr1	24254	24448	-249226173	+	L2b	LINE	3213	3425	-1
(b)	chr1	140784	141290	-249109331	+	MER21C	LTR	26	527	-411
	chr1	141290	141597	-249109024	-	AluJb	SINE	-18	294	2
	chr1	141597	141667	-249108954	+	MER21C	LTR	528	605	-333
	chr1	141667	141970	-249108651	+	AluJr	SINE	1	302	-10
	chr1	141970	142271	-249108350	+	MER21C	LTR	606	919	-19
(c)	chr1	389450	389591	-248861030	+	L1ME3D	LINE	3222	3368	-2778
	chr1	389589	391571	-248859050	+	L1MA8	LINE	4080	6108	-183
	chr1	391571	392307	-248858314	-	L1MA2	LINE	-1	6303	5556
	chr1	392307	392431	-248858190	+	L1MA8	LINE	6109	6238	-53
	chr1	392465	393206	-248857415	+	L1ME3D	LINE	3352	4119	-2027

interruptions occur from left to right sequentially, they move lower and lower down in the parse tree. Therefore, the parse trees are not an accurate reflection of the independent nature of the interruptions, even though that information is encoded in the tree. In this subsection, we will address this situation, by giving real examples and proposing a modification to turn the parse trees into evolutionary trees of interruptions, in order to capture the TE evolution more accurately.

Example 7 illustrates three atomic patterns of interruptions: a single interruption, sequential interruptions and recursive interruptions. These three patterns can exist by themselves, nest with themselves, or mix with other pattern(s) to form more complex interruptions in a genomic sequence. Instead of representing interruptions using a parse tree strictly following the grammar in Definition 11, as in Example 6, a simplified form of trees are used in Example 7, showing these interruptions essentially in the same way of Example 6.

Example 7 Table 6 is a list of TE fragments from chromosome 1 taken from the RepeatMasker TE fragments set, $\bar{\chi}(s \overset{\text{RM}}{\longleftrightarrow} \chi_s)$. The fragments are grouped into three interruptions sets marked as (a), (b) and (c), corresponding to the trees in Fig. 10 (a), (b) and (c), where instead of orders of TEs, the nodes of the trees are labelled with TE names of these fragments.

- Group (a) is a single interruption, where an instance of *MIR* inserted itself into an instance of *L2b*;
- Group (b) shows two sequential interruptions (similar to Example 3), where an instance of *AluJb* and an instance of *AluJr* inserted themselves into an instance of *MER*21*C* and broke *MER*21*C* into three fragments;
- Group (c) shows two recursive interruptions (similar to Example 5), where an instance of *L*1*MA*8 inserted itself into an instance of *L*1*ME*3*D*, then at a later time, an instance of *L*1*MA*2 inserted itself into an instance of *L*1*MA*8.

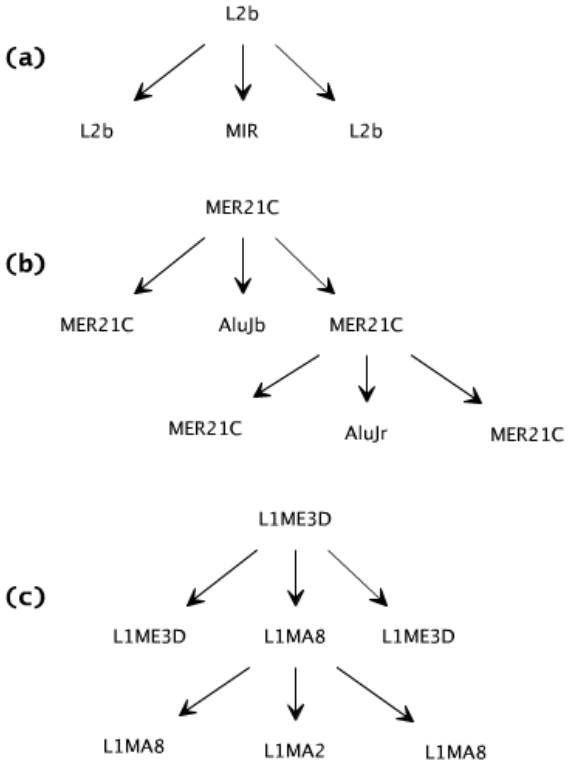

Fig. 10 The corresponding trees in Table 6. (a) is tree of a single interruption; (b) is a tree of two sequential interruptions following the productions of type (3) of the grammar in Definition 11; (c) is a tree of two recursive interruptions following the productions of type (3) of the grammar in Definition 11.

For a simple interruption, the root node of the ordered tree represents the interruptee and the children of that node correspond to the fragments of the interruption and their order in the genomic sequence, which are the left fragment of the interruptee, the interrupter and the right fragment of

the interruptee. The interruption shown in Table 6 group (a) corresponds to the TE fragments tree in Fig. 10 (a), where the root node is the interruptee, labelled as the name of the TE fragment to which it belongs, *L2b*, and the three children of the root are (from left to right) the left fragment of the interruptee, labelled as *L2b*, the interrupter, labelled as *MIR*, and the right fragment of the interruptee, labelled as *L2b*.

Nested interruptions correspond to higher level TE fragments trees following the same rule, as in Fig. 10 (b) and (c).

Notice that, the trees in Figure 10 capture not only the nested TEs by the levels of the tree, but also the orders of the TE fragments within a genomic sequence. An order pruned sequence of the genomic sequence can be generated by traversing the leaves of the tree (and mapping TE names with their orders in the TE set), from the left to right of the tree. They contain all the fragments of interruptions as branches. Nevertheless, from the perspective of the relationship between interrupters and interruptees, some of the branches are redundant, such as the left and right fragments of the interruptees, because the interruptee already appears as the parent node. Moreover, the two sequential interruptions in group (2) are split into two levels in Figure 10 (b), however, this is simply a side-effect of the structure of the parse trees, and the rules of the grammar. They are actually independent. In addition, since we are only interested in the phylogeny of TEs, the positional order does not matter in this case, thus, an ordered tree is not necessary. Therefore, we can simplify the tree, by turning it into an unordered tree, removing the redundant branches, and correcting the level split of the sequential interruptions.

Example 8 shows how to turn the trees in Example 7 into another simplified form of trees, where the redundant branches are removed and the sequential interruptions are moved up into the same level.

Example 8 Given a tree of interruptions as in Example 7, we first turn it into an unordered tree, then there are two steps to simplify the tree:

Step 1: "bring up" the sequential interruptions of the same interruptee to the same level of the tree. This includes the interrupters and the left and right fragments of the interruptee, ;

Step 2: remove all leaves that represent the left and right fragments of interruptees.

Fig. 11 shows an example of simplifying a tree of sequential interruptions in Example 7 in two steps. We use the TE orders, instead of TE names, to label the nodes in this example, as it is more clear. The nodes representing interrupters of the sequential interruptions are coded in pink in the diagram.

Fig. 12 shows two more examples of simplifying the trees using the steps described above. These two trees are mixed patterns of the three atomic patterns in Example 7. As with Fig. 11, we use the TE orders, instead of TE names, to label the nodes, and the nodes representing interrupters of the sequential interruptions are coded in pink in the diagram.

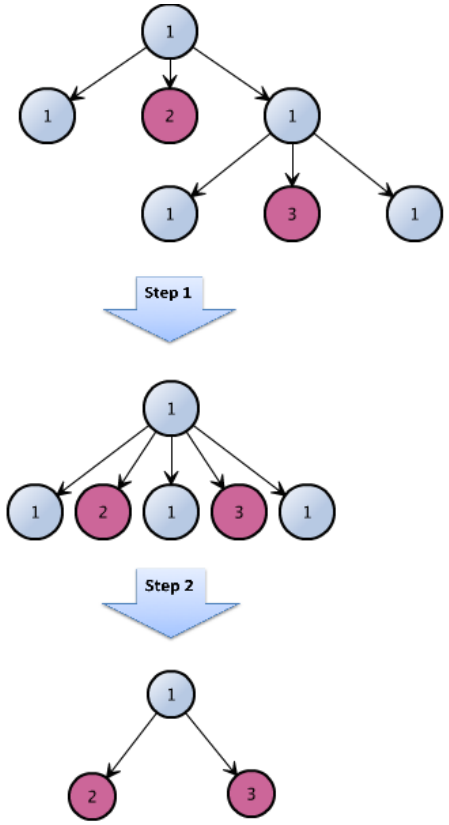

Fig. 11 An example of converting the context-free grammar parse tree to a more standard evolutionary tree for the atomic sequential interruptions pattern.

References

1. Batzer, M., Deininger, P., Hellmann-Blumberg, U., Jurka, J., Labuda, D., Rubin, C., Schmid, C., Zietkiewicz, E., Zuckerkandl, E.: Standardized Nomenclature for Alu Repeats. Journal of Molecular Evolution **42**(1), 3–6 (1996)
2. Belancio, V., Roy-Engel, A., Deininger, P.: All y'all need to know 'bout retroelements in cancer. In: Seminars in Cancer Biology, vol. 20, pp. 200–210. Elsevier (2010)
3. Castilla Valdez, G., Bastiani Medina, S.S.: Iterated local search for the linear ordering problem. International Journal of Combinatorial Optimization Problems & Informatics **3**(1) (2012)
4. Charon, I., Hudry, O.: A branch-and-bound algorithm to solve the linear ordering problem for weighted tournaments. Discrete Applied Mathematics **154**(15), 2097–2116 (2006)
5. DECANI, J.S.: A branch and bound algorithm for maximum likelihood paired comparison ranking. Biometrika **59**(1), 131–135 (1972)
6. Durbin, R., Eddy, S., Krogh, A., Mitchison, G.: Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids. Cambridge University Press, Cambridge (1998)

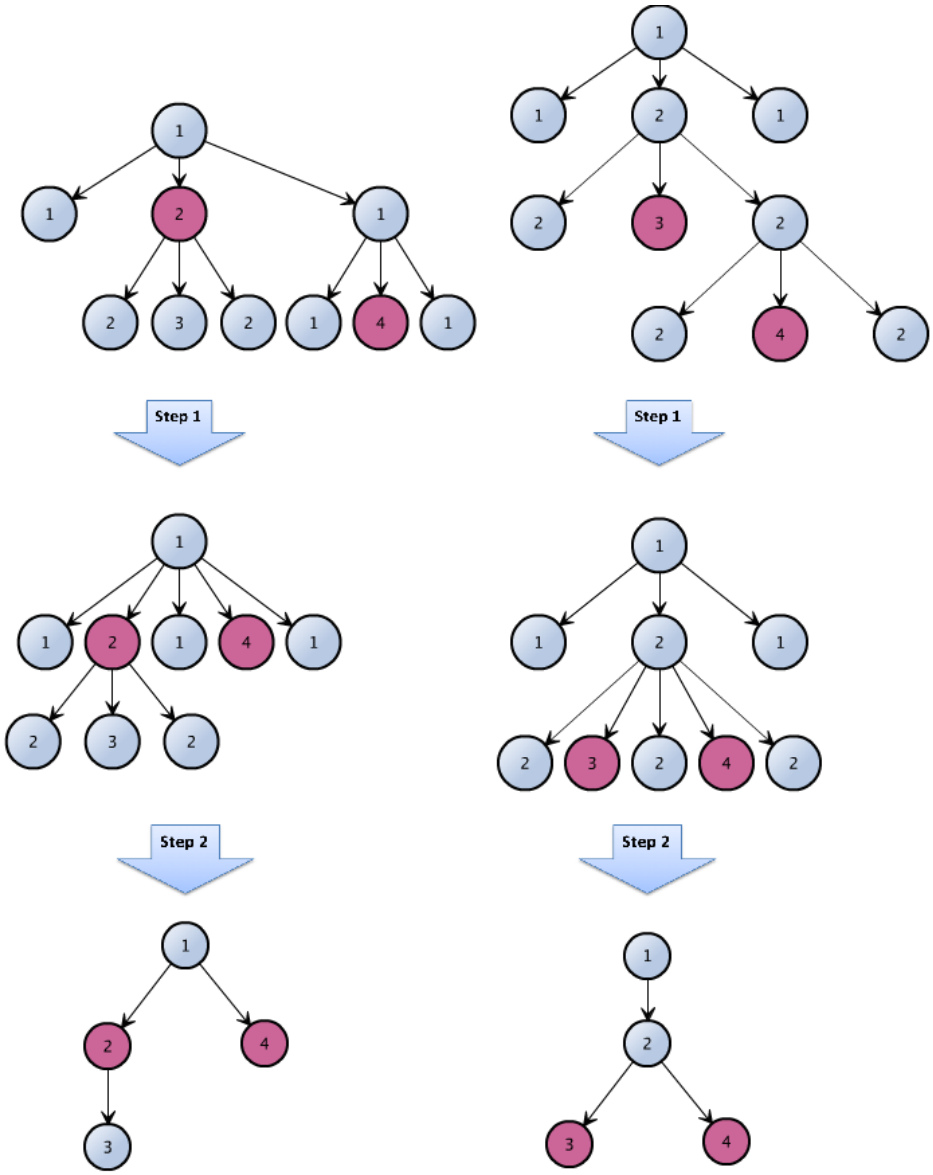

Fig. 12 Two examples of converting the context-free grammar parse trees to standard evolutionary trees for mixed interruption patters.

7. Feo, T.A., Resende, M.G.: Greedy randomized adaptive search procedures. Journal of Global Optimization **6**(2), 109–133 (1995)
8. Finnegan, D.J.: Eukaryotic Transposable Elements and Genome Evolution. Trends in Genetics **5**, 103–107 (1989)
9. Garey, M., Johnson, D.: Computers and Intractability: A Guide to the Theory of NP Completeness, vol. 174. Freeman San Francisco, CA (1979)
10. Giordano, J., Ge, Y., Gelfand, Y., Abrusán, G., Benson, G., Warburton, P.: Evolutionary History of Mammalian Transposons Determined by Genome-wide Defragmentation. PLoS Computational Biology **3**(7), e137 (2007)
11. Glover, F., Laguna, M.: Tabu Search*. Springer (2013)
12. Gregory, T.: The Evolution of the Genome. Academic Press (2005)
13. Hansen, P., Mladenović, N.: Variable Neighborhood Search. Springer (2003)
14. Hopcroft, J.E.: Introduction to Automata Theory, Languages, and Computation, 3/E. Pearson Education India (2008)
15. Jones, N.C., Pevzner, P.: An Introduction to Bioinformatics Algorithms. MIT press (2004)
16. Jurka, J., Kapitonov, V., Pavlicek, A., Klonowski, P., Kohany, O., Walichiewicz, J.: Repbase Update, a database of eukaryotic repetitive elements. Cytogenetic and Genome Research **110**(1-4), 462–467 (2005)
17. Kapitonov, V., Jurkal, J.: The Age of Alu Subfamilies. Journal of Molecular Evolution **42**(1), 59–65 (1996)
18. Kirkpatrick, S., Vecchi, M., et al.: Optimization by simmulated annealing. Science **220**(4598), 671–680 (1983)
19. de Koning, A.J., Gu, W., Castoe, T.A., Batzer, M.A., Pollock, D.D.: Repetitive elements may comprise over two-thirds of the human genome. PLoS Genetics **7**(12), e1002,384 (2011)
20. Korte, B., Oberhofer, W.: Zwei algorithmen zur lösung eines komplexen reihenfolgeproblems. Unternehmensforschung Operations Research-Recherche Opérationnelle **12**, 217–231 (1968)
21. Korte, B., Oberhofer, W.: Triangularizing input-output matrices and the structure of production. European Economic Review **1**(4),

482–511 (1970)
22. Kronmiller, B.A., Wise, R.P.: TEnest: Automated Chronological Annotation and Visualization of Nested Plant Transposable Elements. Plant Physiology **146**(1), 45–59 (2008)
23. Laguna, M., Marti, R., Martí, R.C.: Scatter search: methodology and implementations in C, vol. 24. Springer (2003)
24. Lander, E., Linton, L., Birren, B., Nusbaum, C., Zody, M., Baldwin, J., Devon, K., Dewar, K., Doyle, M., FitzHugh, W., et al.: Initial Sequencing and Analysis of the Human Genome. Nature **409**(6822), 860–921 (2001)
25. Lerat, E.: Identifying Repeats and Transposable Elements in Sequenced Genomes: How to Find Your Way through the Dense Forest of Programs. Heredity **104**(6), 520–533 (2009)
26. Martí, R., Reinelt, G.: The Linear Ordering Problem: Exact and Heuristic Methods in Combinatorial Optimization, vol. 175. Springer (2011)
27. Martí, R., Reinelt, G., Duarte, A.: A benchmark library and a comparison of heuristic methods for the linear ordering problem. Computational Optimization and Applications **51**(3), 1297–1317 (2012)
28. McClintock, B.: Chromosome Organization and Genic Expression. In: Cold Spring Harbor Symposia on Quantitative Biology, vol. 16, pp. 13–47. Cold Spring Harbor Laboratory Press (1951)
29. Schiavinotto, T., Stützle, T.: The linear ordering problem: Instances, search space analysis and algorithms. Journal of Mathematical Modelling and Algorithms **3**(4), 367–402 (2004)
30. Smit, A., Toth, G., Riggs, A., Jurka, J.: Ancestral, Mammalian-wide Subfamilies of LINE-1 Repetitive Sequences. Journal of Molecular Biology **246**(3), 401–417 (1995)
31. Smit A.F.A., H.R., P., G.: RepeatMasker Open-3.0 (1996-2010). Http://www.repeatmasker.org
32. Wicker, T., Sabot, F., Hua-Van, A., Bennetzen, J.L., Capy, P., Chalhoub, B., Flavell, A., Leroy, P., Morgante, M., Panaud, O., et al.: A Unified Classification System for Eukaryotic Transposable Elements. Nature Reviews Genetics **8**(12), 973–982 (2007)