

Sequence analysis

Domain organization within repeated DNA sequences: application to the study of a family of transposable elements

Sébastien Tempel^{1,2}, Mathieu Giraud¹, Dominique Lavenier¹, Israël-César Lerman¹, Anne-Sophie Valin¹, Ivan Couée², Abdelhak El Amrani² and Jacques Nicolas^{1,*}¹IRISA-INRIA, Campus de Beaulieu Bât 12, 35042 Rennes cedex, France and ²CNRS, Université de Rennes 1, UMR 6553 Ecobio, Campus de Beaulieu Bât 14A, 35042 Rennes cedex, France

Received on April 19, 2006; revised on June 9, 2006; accepted on June 15, 2006

Advance Access publication June 29, 2006

Associate Editor: Martin Bishop

ABSTRACT

Motivation: The analysis of repeated elements in genomes is a fascinating domain of research that is lacking relevant tools for transposable elements (TEs), the most complex ones. The dynamics of TEs, which provides the main mechanism of mutation in some genomes, is an essential component of genome evolution. In this study we introduce a new concept of domain, a segmentation unit useful for describing the architecture of different copies of TEs. Our method extracts occurrences of a terminus-defined family of TEs, aligns the sequences, finds the domains in the alignment and searches the distribution of each domain in sequences. After a classification step relative to the presence or the absence of domains, the method results in a graphical view of sequences segmented into domains.

Results: Analysis of the new non-autonomous TE AtREP21 in the model plant *Arabidopsis thaliana* reveals copies of very different sizes and various combinations of domains which show the potential of our method.

Availability: DomainOrganizer web page is available at www.irisa.fr/symbiose/DomainOrganizer/

Contact: DomainOrganizer@irisa.fr

1 INTRODUCTION

Repeated sequences are abundant in eukaryotic genomes and, in some cases, represent most of the genome (Kidwell and Lisch, 2001). Many studies show the relationships between a given family of repeat elements and a host genome, but except for phylogeny, few studies systematically analyze the relationships and variations between copies of a given family of repeats.

Transposable elements (TEs) are present in nearly all genomes that have been studied to date. These TEs move or are copied from one genomic location to another (Craig *et al.*, 2002). TEs are characterized and classified on the basis of terminal or sub-terminal remarkable structures or of their protein-coding capacity. TEs that encode the proteins involved in the amplification mechanism are called autonomous. Amplification mechanisms define two classes of TEs (Fig. 1). Class I elements, or retrotransposons, move via an RNA intermediate. Class II elements, or DNA transposons

seem to move via 'cut-and-paste' mechanisms where the DNA element itself is the mobile intermediate (Fig. 1).

However, many families of both classes do not show any coding capacity (Wessler *et al.*, 1995; Feschotte and Mouches, 2000). Except for some families (SINES, RTEs, etc.) deriving from tRNAs (Kramerov and Vassetzky, 2005), such non-autonomous TEs are thought to derive from autonomous TEs by mutation, insertion or deletion of sequences. Currently, most studies fail to characterize exhaustively and to explain these transformations. Particularly at the sequence level, one lacks methods and tools to study the fine 'architecture' of such repeats: non-autonomous transposons are solely defined by their extremities. A noticeable exception is the MOSAIC tool (Andre *et al.*, 2001), which addresses the segmentation of subtelomeric regions of the yeast genome. MOSAIC is based on the computation of a PI index in an alignment of sequences to be segmented. Each distribution of nucleic acids in a column of the alignment corresponds to a specific value of the index. Overall, segments correspond to dense areas with respect to a PI value.

In the present work we develop a more ambitious association of language analysis, optimization and classification tools that allow identification, characterization and graphical representation of the combinations of elementary domains that make up each sequence of a given family. We applied it to the study of a new family of non-autonomous TEs, called AtREP21, in the whole genome of the model plant *Arabidopsis thaliana*. This family has been chosen as an illustration model of the complex internal organization of non-autonomous transposons and of the wide range of possible variations inside the same family. By analogy with proteins (Servant *et al.*, 2002), we call 'domains' the building blocks of repeated sequences. Automatic identification of domains requires a relatively complex procedure that is described below.

2 METHODS

Our method represents a given family of repeated sequences as a combination of domains. In order to obtain these combinations, we associate existing and original algorithms (Fig. 2). First, for the given family of repeats, characterized by their extremities, all occurrences are found and sequences are aligned together. Our algorithm then detects the set of possible boundaries of domains in the multiple alignment and produces a characterization of each potential domain, based on HMM profiles; the final list of domains

*To whom correspondence should be addressed.

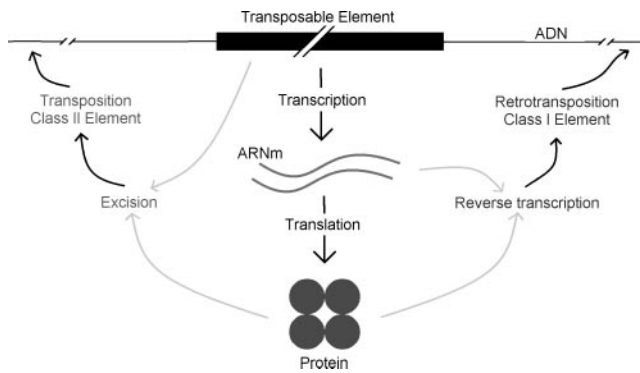


Fig. 1. General mechanisms involved in amplification of autonomous TEs: Autonomous TEs contain protein-encoding genes. Proteins encoded by class 1 TEs are involved in reverse transcription and insertion of the resulting DNA in the host genome. Proteins encoded by class 2 TEs are involved in the excision of TEs and their insertion at another location in the host genome.

results from a distribution analysis of the presence of each domain in each sequence and an extraction of a subset of domains producing an optimal covering of the sequences. This way, each sequence may be segmented with respect to domains. Two final steps allow drawing an overall representation of the family: sequences are classified with respect to the presence or the absence of domains and a visualization tool yields a graphical view of all the sequences segmented into domains (Fig. 2).

2.1 Analysis of the multiple alignment

TEs are a major source of variation in genomes: occurrences of the same repeated sequence may differ greatly in practice, owing to various biological events altering the content of an initial ‘seed’ sequence. Non-autonomous helitrons often present insertions of host sequences within their sequence [nested site, (Brunner *et al.*, 2005)]. In such a case, it becomes hard to reconstruct the autonomous element from the set of non-autonomous sequences. In a preliminary analysis of a few members of the family AtREP21, and in accordance with previous studies of non-autonomous TEs (Kapitonov and Jurka, 2001; Inukai and Sano, 2002), we have clearly observed that some variations conserved across several sequences could be largely ascribed to biological events such as insertion/deletion of mobile DNA or of host sequences. The standard way to analyze such variations is to produce a multiple alignment of repeats from which a consensus sequence may be drawn. Instead of stopping the analysis at this level, with a partial consensual characterization of the family, we propose to fully exploit the alignment in the search of decompositions in elementary domains.

Sequences that do not have the consensual letter at a given position may either correspond to a local mutation with respect to a consensual domain or to the insertion or deletion of another domain. We can measure the variability or ‘heterogeneity’ of sequences in an alignment by various indices. We propose to consider two of them that will be used in our domain detection algorithm and seem characteristic of a variation in the number of possible domains: the first one, NbEmpty, counts the number of sequences that are not represented (due to a gap) in a given window sliding on the alignment; the second one, NbDiff, measures the maximal distance between a sequence and a current consensus sequence. The next section details the use of these indices in our algorithm.

2.2 A new segmentation algorithm from multiple alignment: DomainDetector

The issue of segmenting DNA sequences into meaningful units has been studied since the genomic sequences became available. It has been of

particular interest in the detection of isochores, CpG islands, origin and terminus of replication or coding/non-coding regions (Li *et al.*, 2002). All methods are based on a compositional analysis of sub-sequences, using various statistics. The two main techniques are (1) estimation and segmentation of hidden Markov models, labeling segments with the best model along the chromosome (Samuels *et al.*, 2003) and (2) Recursive binary segmentation, starting from the whole sequence and recursively finding the best split into two sub-sequences with the calculation of two indices, one for the ranking of split choices (e.g. Jensen–Shannon divergence) and one for stopping the recursion (e.g. based on complexity or on a statistical test with respect to random sequences) (Azad *et al.*, 2002; Bernal-Galvan *et al.*, 1996; Li *et al.*, 2002; Oliver *et al.*, 2004). All these studies are looking for a ‘coarse’ decomposition of sequences, based on their global statistical content. However, it is possible to consider the segmentation problem at a finer level, where segments are characterized by the set of words (i.e. sub-sequences) they represent, that is, a language. In both cases, the issue can be stated in a common formal framework and Gionis and Mannila have proposed recently a clear setting of it (Gionis and Mannila, 2003). Analysing the architecture of domains of a sequence may be stated as an optimization problem, they call the (k, h) -segmentation problem consisting in finding the best segmentation of a sequence of length n into k segments, each segment belonging to a set of h sources, with $h \leq k$. This problem is shown to be NP complete under fairly large conditions.

We propose a slightly different setting based on the assumption that sources are sequences that are copied and diverge in genomes, forming identifiable families. So, we start from a family of sequences S , and are looking for a minimal set of subsequences sources D (the domains) such that each element of S may be expressed as a concatenation of elements of D . Since the segmentation takes into account the sequence ordering and several sequences in parallel, it is expected to produce a finer partitioning than a composition-based analysis on a single sequence.

We assume domains to be of size greater than $m = \text{MinSizeDomain}$, which is sufficient to unambiguously characterize each domain. More precisely, we assume that each domain is characterized by a word w of size at least m , so that each occurrence of the domain contains a word that differs from w by at most MaxErrors errors. An error is a substitution, insertion or deletion at a given position in the multiple alignment. It may well be the case that no occurrence matches w exactly.

Our algorithm is based first on a multiple alignment of the set of sequences and then on the detection of local extrema of a heterogeneity function.

We propose to build this ‘heterogeneity’ function on two indices NbEmpty(M) and NbDiff(M) which measures whether an alignment M is changing with respect to the number of domains it contains. The most important, NbEmpty(M), is defined as the number of sequences in M represented by gaps (i.e. the number of lines with m ‘-’ characters in M) and the number of such locally unaligned sequences should be constant for the same configuration of domains. NbDiff(M) is defined as the maximum number of positions differing from consensus(M) that is observed for words in M . Indeed, it is likely that observed variations correspond to limited mutations and that a single domain exists in an alignment only if the number of conserved positions is high enough for all sequences. consensus(M) is a standard notion of consensus word excluding gapped words (i.e. the word with its i -th letter being the most frequent letter in the list of letters at position i in M minus words with gaps). In this way, spurious domains are avoided and each domain is characterized by a word of size at least m . One must note that NbDiff and MaxErrors are bounded by m and that, for alignments containing insertions of gaps for some sequences, NbDiff equals m . The function is just a coding in base $m + 1$ of the two indices, with a priority given on index NbEmpty. For an alignment M of length m , we get the following:

$$\text{Heterogeneity}(M) = (m + 1) \cdot \text{NbEmpty}(M) + \text{NbDiff}(M).$$

Finally, for a given set of sequences, our algorithm detects boundaries of domains as local extrema of the heterogeneity function such that either NbEmpty changes or NbDiff reaches MaxErrors between two boundaries. For each detection, the set of sequences for each domain is retained. This

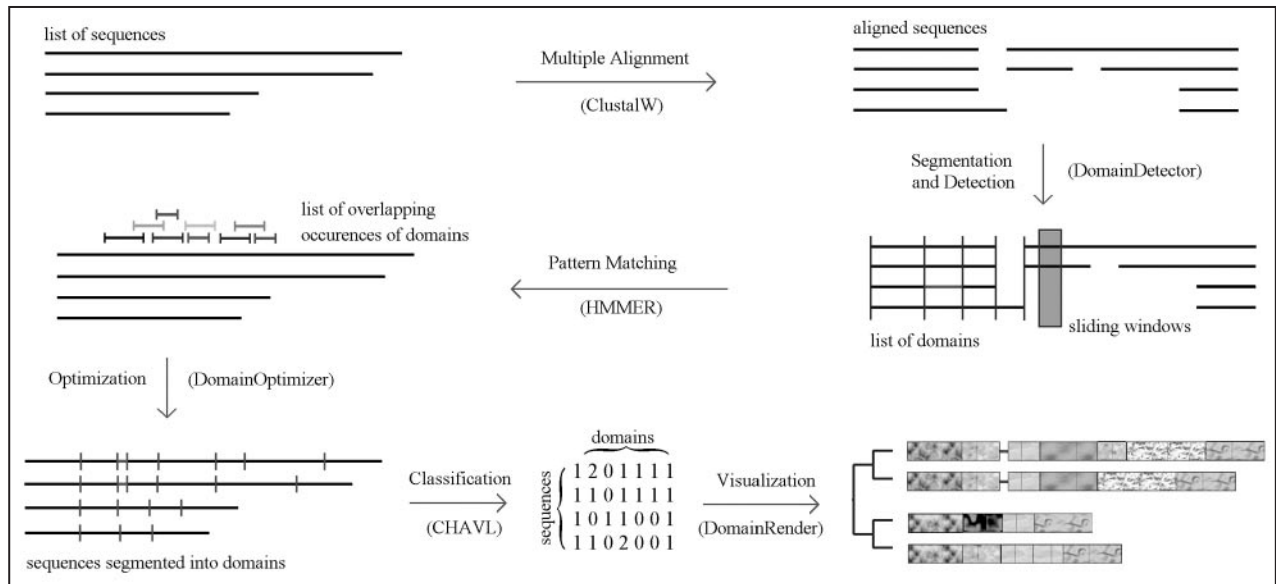


Fig. 2. Main steps of detection and visualization of domains present in the sequences of a given family of repeated DNA. Used programs are indicated in parentheses. New programs are DomainDetector, DomainOptimizer and DomainRenderer. ClustalW (Thompson *et al.*, 1994) creates the multiple alignment of sequences, and DomainDetector finds all potential domains present in this alignment. HMMER is a software suite including HMMbuild and HMMsearch (Eddy, 1998). HMMbuild creates HMM for each detected domain and HMMsearch locates these domains in all the sequences. DomainOptimizer filters a minimum of domains covering all sequences. This detection creates a matrix of presence or absence of domains in the sequences. CHAVL (Lerman, 1993) uses this matrix to create a classification of this family. DomainRenderer creates an image of this family including the classification of sequences and the localization of the domains.

allows building a characteristic description of each domain. More precisely, the algorithm is the following:

```

Proc DomainDetector (Alignment, MinSizeDomain, MaxErrors):
  SeqDomains := ∅; start := 1; m := MinSizeDomain - 1;
  For i := 1 to SizeSequences - m
    Compute (heterogeneity(Alignment[i : i + m]))
  For i := 1 to SizeSequences - m
    If heterogeneity reaches a local extremum and either
      NbEmpty has changed or the sign of |NbDiff-Maxerrors|
      has changed since the previous extremum
      SeqDomains := SeqDomains ∪ ({start, i + m},
        ClassifySeqs (Alignment[start : i + m], MaxErrors));
      start := i + MinSizeDomain
  return SeqDomains
  
```

```

where
ClassifySeqs(M, MaxErrors) = If M = ∅ then return ∅ else
return (S' = {u ∈ M/dist(u, consensus(M)) < MaxErrors}) ∪
ClassifySeqs(MS', MaxErrors)
  
```

We choose a multiple alignment with a low gap insertion and low gap extension cost in order to limit the need for multiple consensus at the same position. This way, different clusters of sequences are naturally shifted with respect to each other in the optimal alignment by the introduction of gaps. Figure 3 shows an example of domain detection in a multiple alignment.

2.3 Characteristic HMM for each domain

A Profile-HMM (Profile hidden Markov model) is a probabilistic model representing an alignment of sequences (Durbin *et al.*, 1998). Each group S' of detected sequences is aligned with ClustalW. HMMER (Eddy, 1998) creates a HMM profile from each alignment and locates all occurrences of HMM profiles, in the complete sequences. Since domains are detected

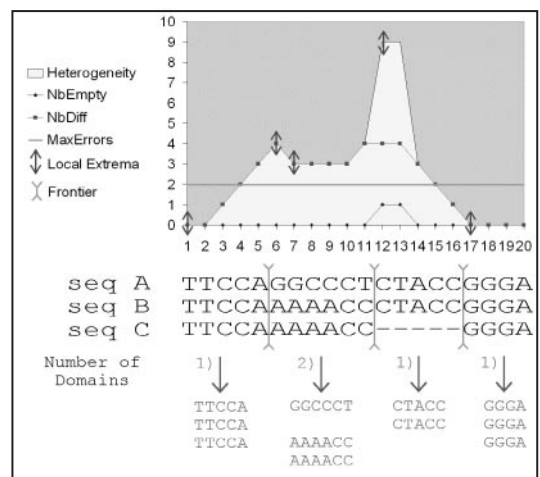


Fig. 3. Example of detection of domains. MinSizeDomain and MaxErrors are set to 4 and 2. The algorithm slides a window of size 4 on the aligned sequence updating at each step NbDiff and NbEmpty. Initially, the number of errors is 0 at position 1. The number of errors increases to a local maximum of 4 errors at position 6. The detected domain corresponds to the alignment from position 1 to $(6 - 1 = 5)$. The window restarts at position $(5 + 4 = 9)$. The window slides until position 12. A sequence gap appears at this position. A new boundary is detected with two domains ranging from position 6–11. The window slides until position 14. At this position, NbEmpty is null and NbDiff equals 2. At the next position the number of errors decreases, the window slides until position 17. At this position, NbDiff and NbEmpty equal 0, this is an extremum. The detected segment lies from 12 to 16. The ending segment [17–20] leads to a last domain.

independently, some larger matches may include or overlap other matches on a DNA sequence. An overlap between two different domains reveals a common motif between these domains but may hardly be interpreted as a new domain flanked by two shorter domains, since generally these potential domains do not occur independently. The sole exception is when the overlapping is almost complete and both HMM profiles match with a high score (E -value in HMMER). In such a case, domains are merged and a new HMM is generated. In the same way, an overlap between two identical domains is characteristic of a 'tandem' overlapping copy, a known pattern in genomes. For large domains, it is also possible to observe included domains, a mobile element being able to integrate another mobile element. Since all these relations observed between domains seem natural, we chose to retain all detected domains instead of considering the intersection of domains.

2.4 Segmenting sequences: DomainOptimizer

Since the previous step may lead to a greater number of domains than necessary and since these domains may overlap in a non-controlled way, we have to introduce a last optimization step. It aims at minimizing the distance between successive domain segments and minimizing the global number of domains. Given a set of occurrences of domains on a set of sequences, this problem is clearly NP-complete. We have chosen a simple greedy approach that proceeds sequentially on the list of sequences, propagating at each step chosen domains. For each sequence, it looks in a combinatorial way for a minimum of the following cost function, with size of domains previously seen set at 0.

$$\sum_{\text{Occurrences}} \text{Distance inter-occurrences} + \sum_{\text{Domains}} \text{Domainsize}$$

Since choices made in the first sequence are crucial, a further minimization step is achieved on all possible choices for the first sequence. DomainOptimizer contributes to a good robustness of the detection with respect to tuning parameters. Note that distance is an absolute value that may correspond to positive or 'negative' gaps (in case of overlapping domains).

3 IMPLEMENTATION

The domain detection program is written in Perl, as well as the main program chaining the tools. The 1.83 version of ClustalW (Thompson *et al.*, 1994) produced the multiple alignment. The 2.3.2 version of HMMer (Eddy, 1998) has been used to build and detect the HMM profiles of domains. The *A.thaliana* genome sequence came from version 03/17 2004 on TAIR website (www.arabidopsis.org). The final classification of sequences has been achieved with Classification Hiérarchique par Analyse de la Vraisemblance des Liens (CHAVL). CHAVL (Lerman *et al.*, 1993) is a software sustaining a very general method of data hierarchical classification called the Likelihood Linkage Analysis method (Lerman 1993).

Our data table is an incidence data table whose entries are positive integers (coding presence and number of occurrences of a domain in a sequence) or 0 (coding absence) (Fig. 2).

The user may tune some parameters in the Perl script: the size of MinSizeDomain and MaxErrors, the cost of opening a gap and extending a gap and the maximal significant threshold where HMM profile hits are considered in the sequences.

3.1 Visualization of internal sequences

A new tool, DomainRender, has been developed to visualize and further analyze a classified set of sequences with their domains. We chose to display domains in left to right and then large to small ordering. This way, included domains are always displayed on the

larger domain background. The program is written in Python and uses a modified version of a drawing library by Fedor Baart and Hans de Wit (<http://www2.sfk.nl/svg/>). DomainRender imports a tree structure from CHAVL to arrange the sequences accordingly and to display the tree beside the domain view. DomainRender outputs results in SVG format, an XML-based Scalable Vector Graphics language that allows modification with any Scalar Vector Graphics-enabled image processing tool (Fig. 4).

DomainRender is available from the OUEST-Genopole® platform.

4 RESULTS AND DISCUSSION

4.1 Characterization of a new non-autonomous transposable element AtREP21

A novel category of DNA transposons, called helitrons, consisting of various autonomous and non-autonomous families, has recently been described in plants and in other eukaryotes (Kapitonov and Jurka, 2001). After an in-depth study of sequences of one of this family, AtREP3, in the genome of the model plant *A.thaliana*, we have discovered an insertion of a non-autonomous helitron in one of its occurrences. This helitron has termini that are not described in the bank of helitrons RepBase (www.girinst.org). We have called this new family AtREP21 in accordance with the classification of Kapitonov and Jurka (Kapitonov and Jurka, 2001). Like all helitrons identified to date, AtREP21 is characterized by terminal and subterminal structures, a TC 5' terminus, a CTRR 3' terminus and a 3' subterminal short hairpin structure. This family has been submitted to and is available in RepBase (Jurka *et al.*, 2005) (<http://www.girinst.org/repbase>).

A preliminary study showed that the optimal size for describing characteristic termini was 36 bp: a shorter sequence was not sufficiently specific, and a longer sequence could not detect copies that have an insertion or deletion near the termini of AtREP21. The study showed the mean size between the two termini to be 500 bp.

We used STAN (Suffix Tree Analyser) (Nicolas *et al.*, 2005) to find all hits of the sequences of interest in the whole genome. STAN searches for SVG-based patterns in genome sequences. SVG (Dong and Searls, 1994; Searls, 1993), a subclass of context-sensitive grammars, are able to model structural features such as repeats, palindromes, stem-loop or pseudo-knots (Searls, 2002). STAN is available on the OUEST-Genopole® platform (www.irisa.fr/symbiose/STAN/). In the SVG language, the grammar was written as follows, accepting nine errors on each element of a motif: TCCCTTTATTATTAAGGGGAAGTACAAATTGAAAT:9- \times (100,1500)-CCGATTGTCCGCGTAAACCGCGGGTAAAACCTAG:9

4.2 Identification of domains in aligned sequences

The above motif found 48 AtREP21 sequences in the whole *A.thaliana* genome. Sequences are numbered following their order of occurrence in the direct strand. These elements range from 315 to 1012 bp. ClustalW made the alignment with a gap opening penalty of 25 and a gap extension penalty of 0.01. MinSizeDomain was set to 26 bp and MaxErrors to 25%. The threshold E -value for merging two domains' models in the AtREP21 sequences with HMMER was set to 10^{-5} . The domain-identification algorithm discovered 37 boundaries, 121 domains before optimization and finally 76 domains in sequences with a global score of 5811.

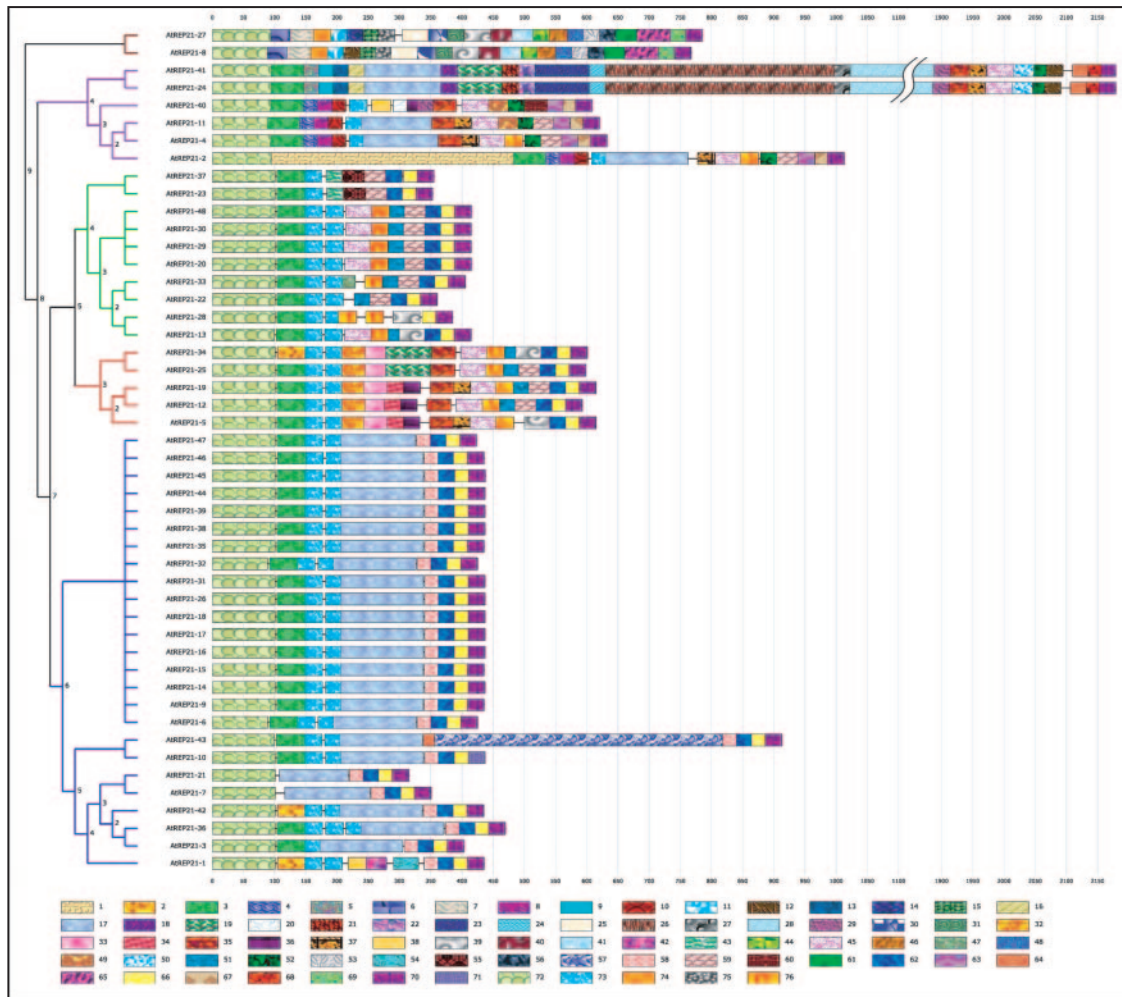


Fig 4. Visualization of domains and CHAVL classification of AtREP21 family. DomainRender draws sequences as a combination of domains.

In order to check the precision of our greedy algorithm, we have computed the minimum score over all possible permutations of the first two elements of the list of sequences (considering two elements instead of one requires 48 times more computations). This could lead to a very different value for the global score and number of domains. A minimum is obtained with a score of 5599 and 73 domains, a low variation with respect to the value we found. The empirical average number of domains is 76 with a SD of 3 and the score 5811 is amongst the 0.6% best scores.

4.3 Visualization and organization of sequences

All domains and their positions were interpreted and displayed with DomainRender. The 5' and 3' terminus, characteristic of AtREP21 family are described by domain number 70 and 72 respectively.

The AtREP21 family shows a significant variation in internal sequences (Fig. 4). This variation of sequences results from insertions, deletions or substitutions of domains. The visualization shows that AtREP21 is mainly divided into five groups of sequences. The first group (color blue in Fig. 4) is composed of AtREP21 number 1, 3–4, 7, 9–10, 14–18, 23, 31–32, 35–36, 38–39, 42–47. This group is mainly composed of the combination of domains n° 2 or 3, 17, 58,

62, 66, 70, 72 and 73, except AtREP21 n° 43, which presents a large insertion of domain 57.

The second group (color red) contains AtREP21 number 5, 12, 19, 25 and 34. It is mainly characterized by the substitution of domains number 17 and 58 with domains number 32–37, 45, 51, 59 and 76 or others domains.

The third group (color green), which contains AtREP21 number 13, 20, 22, 23, 28–30, 33, 37 and 48, is mainly characterized by the substitution of domains 17 and 58 by domains 45, 51, 59 and 76.

The fourth group (color purple) is divided into two subgroups. The first subgroup contains AtREP21 number 2, 4, 11, 40 and is mainly characterized by the insertion of domains 4, 8 and 10 and the substitution of domains number 58, 62, 66, by domains 35, 37, 45, 52, 59, 63, 67 and 76 (AtREP21 n° 2 has a supplementary insertion of domain 1). The last subgroup (AtREP21 n° 24 and 41) is mainly characterized by a large insertion of domains.

The last group is a heterogeneous group containing the remaining sequences of AtREP21 family, characterized by many insertions and deletions of domains.

Our method therefore reveals a complexity and variance of internal structure between members of a given family of TEs.

These differences, which may result in a certain level of disconnection between TE repetition and domain repetition within a genome, is usually not observable by standard sequence analysis tools on DNA such as BLAST (Altschul *et al.*, 1997), or specialized softwares for the analysis of TEs [Recon (Price *et al.*, 2005), Pilers (Edgar and Myers, 2005) or RepeatScout (Bao and Eddy, 2002)].

4.4 Strengths and limitations

A main limit of the method is the necessity of a multiple alignment, currently created by ClustalW, where the gap formed in the multiple alignment is not always optimal. Some sequences, which do not possess an inserted sequence but have a similarity with an inserted sequence, are thus split in many partial sequences. This similarity is the result of a small number of mismatching positions in the alignment. In such a case, DomainDetector tends to fragment domains. We are currently studying whether other alignment methods may be more suited in this context.

Other limits come from parameter tuning. Although we use only two parameters, it is clear that they influence the number and size of detected domains. If the size of minimal domains is too small, the number of domains may be simply too large to give an interesting abstraction of sequence. On the contrary, if the size of domains is too large, the number of domains may be too restricted to formulate a relevant biological interpretation. An interesting track of research would be to adjust the value of this parameter on the global behavior of the distribution of domains.

The other important parameter is the error threshold. By default, this error threshold is fixed at 25% of the minimal size, but some repeated DNA families may require a different threshold. In fact the last optimization step renders the method resistant to a large range of parameters/values. Variations correspond essentially to domain merging, not to a complete redefinition of domains.

The sole comparable algorithm is MOSAIC (Andre *et al.*, 2001), which uses also a ClustalW alignment and the calculation of a score (PI index) for the segmentation of sequences. Our segmentation method differs by at least three major points: (1) We use only two parameters while MOSAIC uses four (overall frequency, threshold distance, MinSize of domains and relative frequency). (2) MOSAIC does not produce a real segmentation of the sequences but rather an identification of homogeneous parts of alignment. In contrast, our approach includes an identification step and an optimization step that lead to a description of each sequence into domains. (3) Mosaic does not search for repeated domains since it does not try any characterization of segments.

Part of the tools we used may be exchanged in our method. The characterization of domains can be provided for instance by combinatorial pattern discovery algorithms such as PRATT (Jonassen, 1997). However, explicit pattern discovery remains more suited for the discovery of patterns in protein sequences. HMM have proved to be efficient on the issue of DNA segmentation (Peshkin and Gelfand, 1999), but have been used rather with the aim of modeling whole sequences that limits in practice the complexity of models. Thus, HMMER has been applied in combination with RepeatMasker on global recognition of TEs (Juretic *et al.*, 2004), which led to combinatorial problems and did not reveal the fine grain architecture of each TE.

Phylogenetic analysis can replace our method of classification. However, other methods often use all nucleotides rather than domains, and the index in our method offers a finer discrimination

of sequences. Given a data table T crossing a set of individuals and a set of binary attributes, consider the matrix S associating to each pair $\{x, y\}$ of individuals the number $s(x, y)$ of attributes common to x and y . If perfect phylogeny can be derived from T , then the matrix S is ultrametric and can be viewed via an ultrametric classification tree. The parsimony methods work on an ordered sequence of character state transformations with the aim to minimize the number of transformations. With this process, implicitly, the similarity index taken into account is the above $s(x, y)$. This index is considered in CHAVL as a raw index from which is built a statistically standardized index with respect to its empirical distribution on pairs of individuals. This way, non-significant effects are neutralized. Furthermore, the final version of the index used for aggregation refers to a probabilistic scale, i.e. to the likelihood of observing the value of the previous index.

Overall, our method shows that a family of repeated DNA sequences can be described and classified at a macroscopic level of internal building blocks of sequences that we call domains. CHAVL, which uses the distribution matrix of domains for classifying the sequences, seems therefore more adequate for our method of domain analysis.

5 CONCLUSION

Our analysis provides structural results on the internal organization of a family of DNA sequences. It can describe the differences between family members in terms of domains content and highlights the evolution of the host genome with respect to these components. This structural and descriptive analysis must be associated with biological knowledge and further analysis of transposition mechanisms and TE-genome relationships. A recent study (Rouleux-Bonnin *et al.*, 2005) shows that the conservation or deletion of a piece of sequence in Mariner-like elements may be associated to its secondary structure. It seems thus relevant to analyze a combination of domains in terms of stable/unstable secondary structure of domains. On the other hand, since TEs are known to be nesting sites for other TEs or for micro- and mini-satellites (Kapitonov and Jurka, 2001; Inukai and Sano, 2002), domains that appear to be unique insertions can be further studied in terms of similarity with previously described TEs or micro- and mini-satellites. Further analysis of secondary structures, such as palindromes, hairpins or inverted repeats at the extremities can also result in the characterization of domains or groups of domains themselves as members of other classes of TEs or satellites. In such cases, the identification of repeated occurrences of these domains or groups of domains outside the initial TE family should be expected. A syntactical approach searching for such structural features seems very relevant for further studies. We have already described the role of Stan (Nicolas *et al.*, 2005) in the initial characterization of a TE family. Other recent approaches confirm the interest of exploiting a structural model of such elements (McCarthy and McDonald, 2003).

ACKNOWLEDGEMENTS

This work was supported in part by a fellowship from the French Ministry of National Education and Research (Ministère de l'éducation nationale et de la recherche) and in part by a grant from région Bretagne, and used the bioinformatics platform of OUEST-Genopole® (<http://genouest.org>). Funding to pay the Open

Access publication charges for this article was provided by INRIA, via a contract with the Réseau National des Génomiques.

Conflict of Interest: none declared.

REFERENCES

- Altschul,S.F. et al. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.
- Andre,C. et al. (2001) MOSAIC: segmenting multiple aligned DNA sequences. *Bioinformatics*, **17**, 196–197.
- Azad,R.K. et al. (2002) Simplifying the mosaic description of DNA sequences. *Phys. Rev. E Stat. Nonlin. Soft Matter Phys.*, **66**, 031913.
- Bao,Z. and Eddy,S.R. (2002) Automated *de novo* identification of repeat sequence families in sequenced genomes. *Genome Res.*, **12**, 1269–1276.
- Bernaola-Galvan,P. et al. (1996) Compositional segmentation and long-range fractal correlations in DNA sequences. *Phys. Rev. E Stat. Phys. Plasmas Fluids Relat. Interdiscip. Topics*, **53**, 5181–5189.
- Brunner,S. et al. (2005) Origins, genetic organization and transcription of a family of non-autonomous helitron elements in maize. *Plant J.*, **43**, 799–810.
- Craig,N.L., Craigie,R., Gellert,M. and Lambowitz,A. (2002) *Mobile DNA II*. American Society for Microbiology Press, Washington DC.
- Dong,S. and Searls,D. (1994) Gene structure prediction by linguistic methods. *Genomics*, **23**, 540–551.
- Durbin,R., Eddy,S.R., Krogh,A. and Mitchison,G. (1998) *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, Cambridge, UK.
- Eddy,S.R. (1998) Profile hidden markov models. *Bioinformatics*, **14**, 755–763.
- Edgar,R.C. and Myers,E.W. (2005) PILER: identification and classification of genomic repeats. *Bioinformatics*, **21** (Suppl. 1), i152–i158.
- Feschotte,C. and Mouches,C. (2000) Evidence that a family of miniature inverted-repeat transposable elements (MITEs) from the *Arabidopsis thaliana* genome has arisen from a pogo-like DNA transposon. *Mol. Biol. Evol.*, **17**, 730–737.
- Gionis,A. and Mannila,H. (2003) Finding Recurrent Sources in Sequences. In *Proceedings of the 7th International Conference on Research in Computational Molecular Biology (RECOMB)*, April 10–13, Berlin, Germany, pp. 123–130.
- Inukai,T. and Sano,Y. (2002) Sequence rearrangement in the AT-rich minisatellite of the novel rice transposable element Basha. *Genome*, **45**, 493–502.
- Jonassen,I. (1997) Efficient discovery of conserved patterns using a pattern graph. *Comput. Appl. Biosci.*, **13**, 509–522.
- Juretic,N. et al. (2004) Transposable element annotation of the rice genome. *Bioinformatics*, **20**, 155–160.
- Jurka,J. et al. (2005) Repbase Update, a database of eukaryotic repetitive elements. *Cytogenet. Genome Res.*, **110**, 462–467.
- Kalyanaraman,A. and Aluru,S. (2005) Efficient algorithms and software for detection of full-length LTR retrotransposons. *Proc. IEEE Comput. Syst. Bioinform. Conf.*, 56–64.
- Kapitonov,V. and Jurka,J. (2001) Rolling-circle transposons in eukaryotes. *Proc. Natl Acad. Sci. USA*, **98**, 8714–8719.
- Kidwell,M.G. and Lisch,D.R. (2001) Perspective: transposable elements and host genome evolution. *Trends Ecol. Evol.*, **15**, 95–99.
- Kramerov,D.A. and Vassetzky,N.S. (2005) Short retroposons in eukaryotic genomes. *Int. Rev. Cytol.*, **247**, 165–221.
- Lerman,I.C. (1993) Likelihood linkage analysis (LLA) classification method; an example treated by hand. *Biochimie*, **75**, 379–397.
- Lerman,I.C. et al. (1993) Principes et calculs de la méthode implantée dans le programme CHAVL (Classification Hiérarchique par Analyse de la Vraisemblance des Liens). Modulad. pp. 33–101.
- Li,W. et al. (2002) Applications of recursive segmentation to the analysis of DNA sequences. *Comput. Chem.*, **26**, 491–510.
- McCarthy,E. and McDonald,J. (2003) LTR_STRUC: a novel search and identification program for LTR retrotransposons. *Bioinformatics*, **19**, 362–367.
- Nicolas,J. et al. (2005) Suffix-tree analyser (STAN): looking for nucleotidic and peptidic patterns in genomes. *Bioinformatics*, **21**, 4408–4410.
- Oliver,J.L. et al. (2004) IsoFinder: computational prediction of isochores in genome sequences. *Nucleic Acids Res.*, **32**, 287–292.
- Peshkin,L. and Gelfand,M. (1999) Segmentation of yeast DNA using hidden Markov models. *Bioinformatics*, **15**, 980–986.
- Price,A.L. et al. (2005) *De novo* identification of repeat families in large genomes. *Bioinformatics*, **21** (Suppl. 1), i351–i358.
- Rouleux-Bonnin,F. et al. (2005) Evolution of full-length and deleted forms of the mariner-like element, Botmar1, in the Genome of the bumble bee, *Bombus terrestris* (Hymenoptera: Apidae). *J. Mol. Evol.*, **60**, 736–747.
- Samuels,D. et al. (2003) A compositional segmentation of the human mitochondrial genome is related to heterogeneities in the guanine mutation rate. *Nucleic Acids Res.*, **31**, 6043–6052.
- Searls,D. (1993) String variable grammar: a logic grammar formalism for the biological language of DNA. *J. Logic Program.*, **12**, 1–30.
- Searls,D. (2002) The language of genes. *Nature*, **420**, 211–217.
- Servant,F. et al. (2002) ProDom: automated clustering of homologous domains. *Brief. Bioinform.*, **3**, 246–251.
- Thompson,J. et al. (1994) ClustalW: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res.*, **22**, 4673–4680.
- Wessler,S. et al. (1995) LTR-retrotransposons and MITES: important players in the evolution of plant genomes. *Genet. Dev.*, **5**, 814–821.